



Migrating from RedHat to SUSE Linux Enterprise Server 10

COURSE 3068

Novell Training Services

www.novell.com

AUTHORIZED COURSEWARE

Proprietary Statement

Copyright © 2006 Novell, Inc. All rights reserved.

No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express prior consent of the publisher. This manual, and any portion thereof, may not be copied without the express written permission of Novell, Inc. Novell, Inc.

1800 South Novell Place
Provo, UT 84606-2399

Disclaimer

Novell, Inc. makes no representations or warranties with respect to the contents or use of this manual, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose.

Further, Novell, Inc. reserves the right to revise this publication and to make changes in its content at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any NetWare software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose.

Further, Novell, Inc. reserves the right to make changes to any and all parts of NetWare software at any time, without obligation to notify any person or entity of such changes.

This Novell Training Manual is published solely to instruct students in the use of Novell networking software. Although third-party application software packages are used in Novell training courses, this is for demonstration purposes only and shall not constitute an endorsement of any of these software applications.

Further, Novell, Inc. does not represent itself as having any particular expertise in these application software packages and any use by students of the same shall be done at the students' own risk.

Software Piracy

Throughout the world, unauthorized duplication of software is subject to both criminal and civil penalties.

If you know of illegal copying of software, contact your local Software Antipiracy Hotline.

For the Hotline number for your area, access Novell's World Wide Web page at <http://www.novell.com> and look for the piracy page under "Programs."

Or, contact Novell's anti-piracy headquarters in the U.S. at 800-PIRATES (747-2837) or 801-861-7101.

Trademarks

Novell, Inc. has attempted to supply trademark information about company names, products, and services mentioned in this manual. The following list of trademarks was derived from various sources.

Novell, Inc. Trademarks

Novell, the Novell logo, NetWare, BorderManager, ConsoleOne, DirXML, GroupWise, iChain, ManageWise, NDPS, NDS, NetMail, Novell Directory Services, Novell iFolder, Novell SecretStore, Ximian, Ximian Evolution and ZENworks are registered trademarks; CDE, Certified Directory Engineer and CNE are registered service marks; eDirectory, Evolution, exteNd, exteNd Composer, exteNd Directory, exteNd Workbench, Mono, NIMS, NLM, NMAS, Novell Certificate Server, Novell Client, Novell Cluster Services, Novell Distributed Print Services, Novell Internet Messaging System, Novell Storage Services, Nsure, Nsure Resources, Nterprise, Nterprise Branch Office, Red Carpet and Red Carpet Enterprise are trademarks; and Certified Novell Administrator, CNA, Certified Novell Engineer, Certified Novell Instructor, CNI, Master CNE, Master CNI, MCNE, MCNI, Novell Education Academic Partner, NEAP, Ngage, Novell Online Training Provider, NOTP and Novell Technical Services are service marks of Novell, Inc. in the United States and other countries. SUSE is a registered trademark of SUSE LINUX GmbH, a Novell company. For more information on Novell trademarks, please visit <http://www.novell.com/company/legal/trademarks/tmlist.html>.

Other Trademarks

Adaptec is a registered trademark of Adaptec, Inc. AMD is a trademark of Advanced Micro Devices. AppleShare and AppleTalk are registered trademarks of Apple Computer, Inc. ARCserv is a registered trademark of Cheyenne Software, Inc. Btrieve is a registered trademark of Pervasive Software, Inc. EtherTalk is a registered trademark of Apple Computer, Inc. Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. Linux is a registered trademark of Linus Torvalds. LocalTalk is a registered trademark of Apple Computer, Inc. Lotus Notes is a registered trademark of Lotus Development Corporation. Macintosh is a registered trademark of Apple Computer, Inc. Netscape Communicator is a trademark of Netscape Communications Corporation. Netscape Navigator is a registered trademark of Netscape Communications Corporation. Pentium is a registered trademark of Intel Corporation. Solaris is a registered trademark of Sun Microsystems, Inc. The Norton AntiVirus is a trademark of Symantec Corporation. TokenTalk is a registered trademark of Apple Computer, Inc. Tru64 is a trademark of Digital Equipment Corp. UnitedLinux is a registered trademark of UnitedLinux. UNIX is a registered trademark of the Open Group. WebSphere is a trademark of International Business Machines Corporation. Windows and Windows NT are registered trademarks of Microsoft Corporation.

All other third-party trademarks are the property of their respective owners.

Contents

Introduction

Course Objectives	Intro-2
Audience	Intro-3
Certification and Prerequisites	Intro-3
SUSE Linux Enterprise Server 10 Support and Maintenance	Intro-5
Novell Customer Center	Intro-6
SUSE Linux Enterprise Server 10 Online Resources	Intro-7
Agenda	Intro-8
Exercises	Intro-8
Exercise Conventions	Intro-9

SECTION 1 **Install SUSE Linux Enterprise Server 10**

Objectives	1-1
Objective 1 Perform a SLES 10 Installation	1-2
Boot from the Installation Media	1-2
Select the System Language	1-5
Select the Installation Mode	1-6
Set the Clock and Time Zone	1-8
Understand and Change the Installation Settings	1-9
Verify Partitioning	1-10
Select Software	1-25
Start the Installation Process	1-28

Objective 2	Configure the SLES 10 Installation	1-29
	Set the Host Name	1-29
	Set the root Password	1-30
	Configure the Network	1-31
	Test the Internet Connection	1-38
	Novell Customer Center Configuration and Online Update	1-39
	Configure Network Services	1-42
	Manage Users	1-43
	Configure Hardware	1-47
	Finalize the Installation Process	1-49
Objective 3	Troubleshoot the Installation Process	1-50
	Exercise 1-1 Install SUSE Linux Enterprise Server 10	1-53
	Summary	1-54
SECTION 2	YaST	
	Objectives	2-1
Objective 1	Get to Know YaST.	2-2
	The YaST User Interface	2-2
	The Role of SuSEconfig	2-5
	Exercise 2-1 Get to Know YaST	2-7
Objective 2	YaST Software Management.	2-8
	Manage Installation Sources	2-8
	Install Software Packages	2-9
	Apply Security Updates	2-12
	Exercise 2-2 Install New Software	2-17
Objective 3	Manage User and Group Accounts with YaST	2-18
	Basics about Users and Groups	2-18
	User and Group Administration with YaST	2-19
	Set Defaults for New User Accounts	2-30
	Configure Security Settings	2-33
	Exercise 2-3 Manage User Accounts with YaST	2-44

	Exercise 2-4 Configure the Password Security Settings	2-44
Objective 4	Find the YaST Module You Need	2-45
	Summary	2-48
SECTION 3	Network Configuration	
	Objectives	3-1
Objective 1	Manage the Network with YaST	3-2
	Exercise 3-1 Manage the Network	
	Configuration Information from YaST.	3-13
Objective 2	Configure the Network Manually	3-14
	Set Up Network Interfaces with the ip Tool	3-14
	Set Up Routing with the ip Tool	3-26
	Exercise 3-2 Configure the Network	
	Connection Manually	3-31
Objective 3	Configure Host Name and Name Resolution.	3-32
	Set the Host and Domain Name	3-32
	Configure Name Resolution	3-32
	Files Holding the Network Configuration	3-33
Objective 4	Use the NetworkManager to Configure the Network	3-35
	Summary	3-38
SECTION 4	Manage the Linux File System	
	Objectives	4-1
Objective 1	Select a Linux File System	4-2
	Linux File Systems	4-2
	Linux File System Internals	4-4
	File System Journaling	4-11
	Additional File System Documentation	4-12

Objective 2	Configure Linux File System Partitions	4-13
	Linux Device and Partition Names	4-13
	Design Guidelines for Implementing Partitions	4-15
	Manage Partitions with YaST	4-18
	Manage Partitions with fdisk	4-20
Objective 3	Manage Linux File Systems.	4-21
	Create a File System with YaST	4-21
	Create a File System with Command Line Tools	4-23
	Mount File Systems	4-25
	Exercise 4-1 Configure Partitions on Your Hard Drive	4-31
	Check a File System	4-32
	Exercise 4-2 Manage File Systems from the Command Line	4-36
Objective 4	Configure Logical Volume Manager (LVM) and Software RAID	4-37
	Use LVM Components	4-37
	Use LVM Features	4-39
	Configure Logical Volumes with YaST	4-40
	Configure LVM with Command Line Tools	4-46
	Manage Software RAID	4-49
	Exercise 4-3 Create Logical Volumes	4-52
	Summary	4-53
SECTION 5	Manage System Initialization	
	Objectives	5-1
Objective 1	Describe the Linux Load Procedure	5-2
	BIOS and Boot Manager	5-3
	Kernel	5-3
	initramfs (Initial RAM File System)	5-4
	init	5-5

Objective 2	GRUB (Grand Unified Bootloader).....	5-7
	What a Boot Manager Is	5-7
	Boot Managers in SUSE Linux	5-8
	Start the GRUB Shell	5-10
	Modify the GRUB Configuration File	5-11
	Configure GRUB with YaST	5-13
	Boot a System Directly into a Shell	5-19
	Exercise 5-1 Manage the Boot Loader	5-22
Objective 3	Manage Runlevels	5-23
	The init Program and Linux Runlevels	5-23
	init Scripts and Runlevel Directories	5-28
	Change the Runlevel	5-41
	Compare Start Scripts between RHEL and SLES 10	5-43
	Exercise 5-2 Manage Runlevels	5-44
	Summary	5-45
 SECTION 6 Configure Mail and Web Services		
	Objectives	6-1
Objective 1	Postfix	6-2
	Understand the Architecture and Components of Postfix	6-2
	Configure Postfix	6-9
	Exercise 6-1 Send Mail in the Local Network	6-26
	Exercise 6-2 Use Postfix on the Internet	6-29
	Exercise 6-3 Use Lookup Tables	6-45
	Use Postfix Tools	6-46
Objective 2	Apache Web Server	6-48
	Setup a Basic Web Server	6-48
	Exercise 6-4 Install Apache	6-52
	Exercise 6-5 Test the Apache Installation	6-52
	Configure Virtual Hosts	6-56
	Exercise 6-6 Configure a Virtual Host	6-60

Summary	6-61
---------------	------

SECTION 7 AppArmor

Objectives	7-1
Objective 1 Understand the Difference between SELinux and AppArmor	7-2
Discretionary Access Control	7-2
Linux Security Modules	7-3
Mandatory Access Control	7-4
Objective 2 Create and Manage AppArmor Profiles	7-7
Understand Profiles and Rules	7-8
Administer AppArmor Profiles with YaST	7-11
Administer AppArmor Profiles with Command Line Tools ..	7-21
Exercise 7-1 AppArmor	7-25
Objective 3 Control AppArmor	7-26
Start and Stop AppArmor	7-26
View AppArmor's Status	7-27
Reload Profiles	7-29
Objective 4 Monitor AppArmor	7-31
Security Event Report	7-31
Security Event Notification	7-34
Summary	7-36

SECTION 8 Manage Virtualization with Xen

Objectives	8-1
Objective 1 Understand the Concept of Virtualization	8-2
Objective 2 Understand How Xen Works	8-3
Understand Virtualization Methods	8-4
Understand the Xen Architecture	8-6

Objective 3	Install Xen	8-8
	Exercise 8-1 Install Xen	8-11
Objective 4	Manage Xen Domains with YaST	8-12
	Exercise 8-2 Install a Guest Domain	8-18
Objective 5	Manage Xen Domains at the Command Line	8-19
	Understand a Domain Configuration File	8-19
	Use the xm Tool	8-21
	Exercise 8-3 Change Memory Allocation of a Guest Domain	8-25
	Automate Domain Startup and Shutdown	8-26
	Exercise 8-4 Automate Domain Startup	8-27
Objective 6	Understand Xen Networking	8-28
	Understand the Basic Networking Concept	8-28
	Understand Bridging	8-29
	Understand the Network Interfaces in domain0	8-30
	Exercise 8-5 Check the Network Configuration	8-34
Objective 7	Migrate a Guest Domain	8-35
	Use Domain Save and Restore	8-35
	Use Migration and Live Migration	8-36
	Summary	8-37
SECTION 9	iSCSI	
	Objectives	9-1
Objective 1	iSCSI Background	9-2
Objective 2	iSCSI Configuration	9-4
	Red Hat Enterprise Linux 4	9-4
	SUSE Linux Enterprise Server 10	9-5
	Exercise 9-1 Set up an iSCSI Target and an iSCSI initiator . .	9-23
	Summary	9-24

SECTION 10 Cluster File Systems

	Objectives	10-1
Objective 1	Global File System (GFS)	10-2
Objective 2	Oracle Cluster File System 2 (OCFS2)	10-3
	OCFS2 Background	10-3
	OCFS2 Configuration	10-4
	OCFS to OCFS2 Migration	10-11
	GFS - OCFS2 Comparison Table	10-12
	Exercise 10-1 Set up an OCFS2	10-13
	Summary	10-14

SECTION A Appendix: AutoYaST

	Objectives	A-1
Objective 1	Autoinstallation Basics	A-2
	Kickstart on RHEL4	A-2
	AutoYaST on SUSE Linux Enterprise Server 10	A-2
Objective 2	Set up an Installation Server	A-5
Objective 3	Create a Configuration File for AutoYaST	A-11
Objective 4	Start the Installation	A-16
	Summary	A-18

**SECTION B Appendix: Migrating Services from RedHat to SUSE
Linux Enterprise Server 10**

 Objectives B-1

 Objective 1 Migrating Services B-2

**SECTION C Appendix: A Guide to SUSE Linux Enterprise Server for
Red Hat Users**

Introduction

Migrating from RedHat to SUSE Linux Enterprise Server 10 (Course 3068) focuses on the main differences between Red Hat Enterprise Linux 4 and SUSE Linux Enterprise Server 10.

It enables an experienced Linux administrator familiar with RedHat Enterprise Linux 4 to perform administrative tasks on the SUSE Linux Enterprise Server 10 platform.

By covering the administrative details specific to SUSE Linux Enterprise Server 10, this course prepares you to take the Novell Certified Linux Professional 10 (Novell CLP 10) certification practicum test—provided you already have the general Linux knowledge covered in the courses *SUSE Linux Enterprise Server 10 Fundamentals* (Course 3071), *SUSE Linux Enterprise Server 10 Administration* (Course 3072), and *SUSE Linux Enterprise Server 10 Advanced Administration* (Course 3073).

This course also covers topics that are not part of the Novell CLP 10 certification, but are of importance in migration scenarios, like iSCSI, OCFS, or specific differences in mail or web server configuration.

The contents of your student kit include the following:

- *Migrating from RedHat to SUSE Linux Enterprise Server 10* Manual
- *Migrating from RedHat to SUSE Linux Enterprise Server 10* Workbook
- *Migrating from RedHat to SUSE Linux Enterprise Server 10* Course DVD
- *SUSE LINUX Enterprise Server 10* Product DVD
- *SUSE LINUX Enterprise Desktop 10* Product DVD

The *Migrating from RedHat to SUSE Linux Enterprise Server 10* Course DVD contains an image of a SUSE Linux Enterprise Server 10 installation that you can use with the *Migrating from RedHat to SUSE Linux Enterprise Server 10* Workbook outside the classroom to practice the skills you need to take the Novell CLP 10 Practicum exam.



Instructions for setting up a self-study environment are in the setup directory on the Course DVD.

Course Objectives

This course teaches theory as well as practical application with hands-on labs of the following *Migrating from Red at to SUSE Linux Enterprise Server 10* topics on SUSE Linux Enterprise Server 10:

- Install SUSE Linux Enterprise Server 10
- YaST
- Network Configuration
- Manage the Linux File System
- Manage System Initialization
- Configure Mail and Web Services
- AppArmor
- Manage Virtualization with Xen
- iSCSI
- Cluster File Systems

The Appendix includes a section on AutoYaST and a section with information on points to watch on actual migrations.

Audience

The primary audience for this course are experienced Red Hat Linux Administrators, like RHCEs, or those with comparable knowledge.

Certification and Prerequisites

This course helps to prepare for the Novell Certified Linux Professional 10 (CLP 10) Practicum Exam, called the *Practicum*. The Novell CLP 10 is a prerequisite for the higher level certification Novell CLE 10 Practicum.

As with all Novell certifications, course work is recommended. To achieve the certification, you are required to pass the Novell CLP 10 Practicum (050-697).

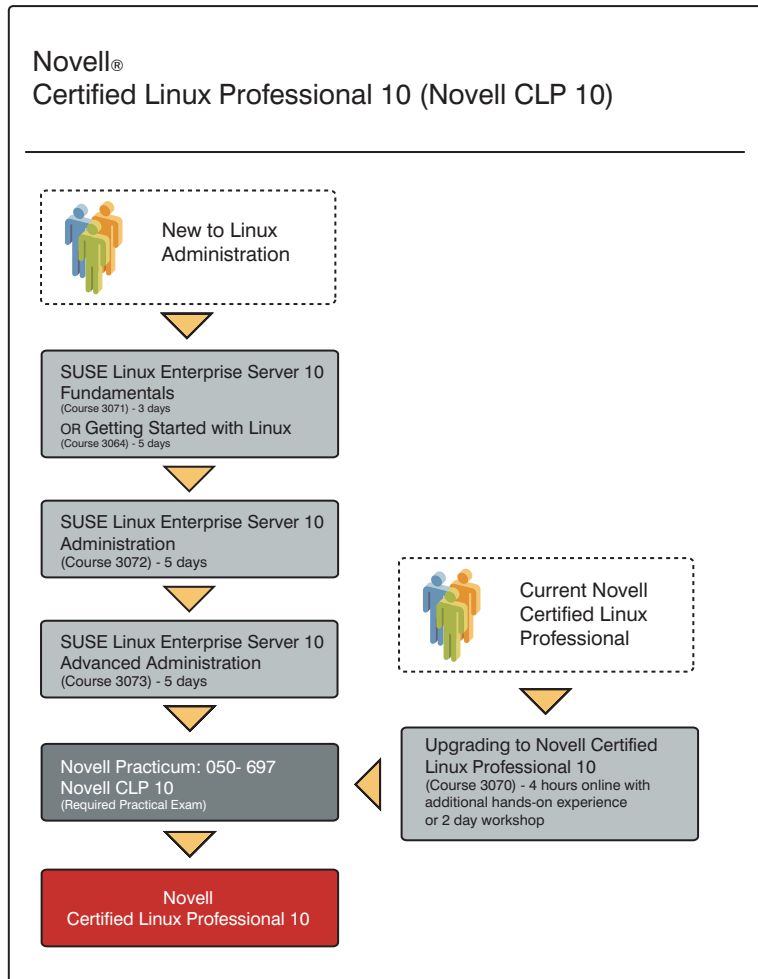
The Novell CLP 10 Practicum is a hands-on, scenario-based exam where you apply the knowledge you have learned to solve real-life problems—demonstrating that you know what to do and how to do it.

The practicum tests you on objectives covered in the courses:

- *SUSE Linux Enterprise Server Fundamentals* (Course 3071)
- *SUSE Linux Enterprise Server Administration* (Course 3072)
- *SUSE Linux Enterprise Server Advanced Administration* (Course 3073)

The following illustrates the training and testing path for Novell CLP 10:

Figure Intro-1





For more information about Novell certification programs and taking the Novell CLP 10 and CLE 10 Practicum exam, see <http://www.novell.com/training/certinfo/>, <http://www.novell.com/training/certinfo/clp10>, and <http://www.novell.com/training/certinfo/cle10>.

Before attending this course, you should have the general Linux knowledge (i.e. the topics not specific to SUSE Linux Enterprise Server 10) covered in the courses:

- *SUSE Linux Enterprise Server 10 Fundamentals* (Course 3071)
- *SUSE Linux Enterprise Server Administration* (Course 3072)
- *SUSE Linux Enterprise Server Advanced Administration* (Course 3073)

SUSE Linux Enterprise Server 10 Support and Maintenance

The copy of SUSE Linux Enterprise Server 10 you received in your student kit is a fully functioning copy of the SUSE Linux Enterprise Server 10 product.

However, to receive official support and maintenance updates, you need to do one of the following:

- Register for a free registration/serial code that provides you with 30 days of support and maintenance.
- Purchase a copy of SUSE Linux Enterprise Server 10 from Novell (or an authorized dealer).

You can obtain your free 30-day support and maintenance code at <http://www.novell.com/products/server/eval.html>.



You will need to have or create a Novell login account to access the 60-day evaluation.

Novell Customer Center

Novell Customer Center is an intuitive, web-based interface that helps you to manage your business and technical interactions with Novell. Novell Customer Center consolidates access to information, tools and services such as:

- Automated registration for new SUSE Linux Enterprise products
- Patches and updates for all shipping Linux products from Novell
- Order history for all Novell products, subscriptions and services
- Entitlement visibility for new SUSE Linux Enterprise products
- Linux subscription-renewal status
- Subscription renewals via partners or Novell

For example, a company might have an administrator who needs to download SUSE Linux Enterprise software updates, a purchaser who wants to review the order history and an IT manager who has to reconcile licensing. With Novell Customer Center, the company can meet all these needs in one location and can give each user access rights appropriate to their roles.

You can access the Novell Customer Center at <http://www.novell.com/center/>.

SUSE Linux Enterprise Server 10 Online Resources

Novell provides a variety of online resources to help you configure and implement SUSE Linux Enterprise Server 10.

These include the following:

- <http://www.novell.com/products/server/>
This is the Novell home page for SUSE Linux Enterprise Server.
- <http://www.novell.com/documentation/sles10/index.html>
This is the Novell Documentation web site for SLES 10.
- <http://support.novell.com/linux/>
This is the home page for all Novell Linux support, and includes links to support options such as the Knowledgebase, downloads, and FAQs.
- <http://www.novell.com/coolsolutions/>
This Novell web site provides the latest implementation guidelines and suggestions from Novell on a variety of products, including SUSE Linux.

Agenda

The following is the agenda for this 3-day course:

TableIntro-1

	Section	Duration
Day 1	Introduction	00:30
	Section 1: Install SUSE Linux Enterprise Server 10	02:00
	Section 2: YaST	02:00
	Section 3: Network Configuration	01:00
Day 2	Section 4: Manage the Linux File System	01:30
	Section 5: Manage System Initialization	01:30
	Section 6: Configure Web and Mail Services	01:30
	Section 7: AppArmor	02:00
Day 3	Section 8: Manage Virtualization with Xen	03:00
	Section 9: iSCSI	01:45
	Section 10: Cluster File Systems	01:45

Exercises

The exercises in this course consist of a description of the exercise, and step-by-step instructions on how to complete the task.

You should first try to complete the task described on you own, based on what is covered in the manual in the respective section. Resort to the step-by-step instruction only if you feel unable to complete the task or to find out if what you did was correct.

The exercises are contained in a separate workbook.

Exercise Conventions

When working through an exercise, you will see conventions that indicate information you need to enter that is specific to your server.

The following describes the most common conventions:

- ***italicized/bolded text***. This is a reference to your unique situation, such as the host name of your server.

For example, if the host name of your server is `da10`, and you see the following:

hostname.digitalairlines.com

you would enter

da10.digitalairlines.com

- ***10.0.0.xx***. This is the IP address that is assigned to your SLES 10 server.

For example, if your IP address is `10.0.0.10`, and you see the following:

10.0.0.xx

you would enter

10.0.0.10

- **Select**. The word *select* is used in exercise steps to indicate a variety of actions including clicking a button on the interface and selecting a menu item.
- **Enter and Type**. The words *enter* and *type* have distinct meanings.

The word *enter* means to type text in a field or at a command line and press the Enter key when necessary. The word *type* means to type text without pressing the Enter key.

If you are directed to type a value, make sure you do not press the Enter key or you might activate a process that you are not ready to start.

SECTION 1 **Install SUSE Linux Enterprise Server 10**

YaST (Yet another Setup Tool) provides options that make installation simple and quick.

However, you also need to understand the more advanced installation options available. By changing installation mode, partitioning, software selection, authentication method, or hardware setup, you can install servers that meet a variety of needs.

In this section, you install SUSE Linux Enterprise Server 10 (SLES 10). You also learn how to use advanced installation options and to troubleshoot the installation process.

Objectives

1. Perform a SLES 10 Installation
2. Configure the SLES 10 Installation
3. Troubleshoot the Installation Process

Objective 1 **Perform a SLES 10 Installation**

Installing SLES 10 consists of a base installation phase and a configuration phase.

To perform the base installation do the following:

- Boot from the Installation Media
- Select the System Language
- Select the Installation Mode
- Set the Clock and Time Zone
- Understand and Change the Installation Settings
- Verify Partitioning
- Select Software
- Start the Installation Process

Boot from the Installation Media

To start the installation process, insert the *SUSE Linux Enterprise Server 10* Product DVD into the DVD drive and then reboot the computer to start the installation program.

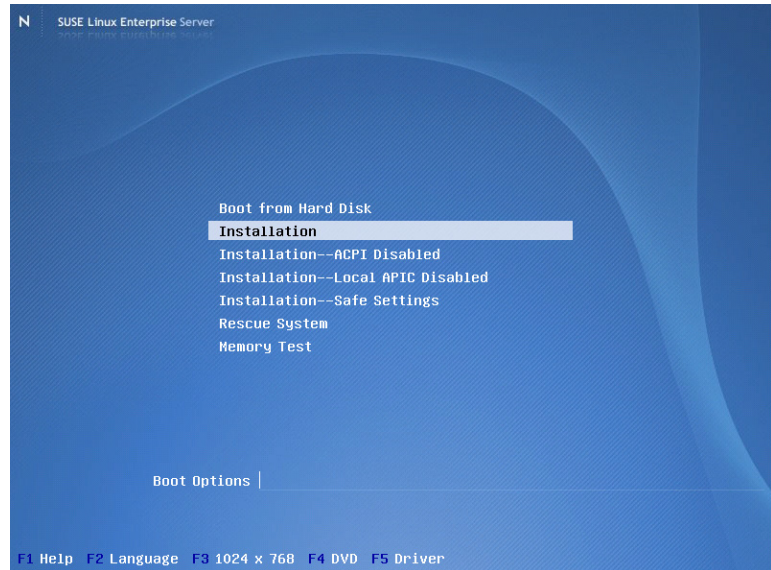


To start the installation program, your computer needs to be configured to start from a DVD drive. You might need to change the boot drive order in the BIOS setup of your system to boot from the drive.

Consult the manual shipped with your hardware for further information.

When your system has started from the installation CD, the following appears:

Figure 1-1



You can use the arrow keys to select one of the following options:

- **Boot from Hard Disk.** Boots the system installed on the hard disk (the system normally booted when the machine is started). This is the default option.
- **Installation.** Starts the normal installation process. All modern hardware functions are enabled.
- **Installation - ACPI Disabled.** Starts the installation process with ACPI (Advanced Configuration and Power Interface) disabled. If the normal installation fails, the reason might be that the system hardware does not support ACPI. In this case, you can use this option to install without ACPI support.

- **Installation - Local APIC Disabled.** Starts the installation process with local APIC (Advanced Programmable Interrupt Controller) disabled.
- **Installation - Safe Settings.** Starts the installation process with the DMA (Direct Memory Access) mode and any interfering power management functions disabled. Use this option if the installation fails with the other options.
- **Rescue System.** Starts the SLES 10 rescue system. If you cannot boot your installed Linux system, you can boot the computer from the DVD (or the first CD if you are using a CD set) and select this option. This starts a minimal Linux system without a graphical user interface to allow experts to access disk partitions for troubleshooting and repairing an installed system.
- **Memory Test.** Starts a memory testing program, which tests system RAM by using repeated read and write cycles. This is done in an endless loop, because memory corruption often shows up sporadically and many read and write cycles might be necessary to detect it.

If you suspect that your RAM might be defective, start this test and let it run for several hours. If no errors are detected, you can assume that the memory is intact. Terminate the test by rebooting the system.

Use the function keys, as indicated in the bar at the bottom of the screen, to change a number of installation settings:

- **F1.** Opens context-sensitive help for the currently selected option of the boot screen.
- **F2.** Select an installation language.
- **F3.** Select a graphical display mode (such as 640x480 or 1024X768) for the installation. You can select one of these, or select text mode, which is useful if the graphical modes cause display problems.

- **F4.** Select an installation media type. Normally you install from the inserted installation disk, but in some cases you might want to select another source, such as FTP or NFS.
- **F5.** Add a driver update CD to the installation process. You are asked to insert the update disk at the appropriate point in the installation process.

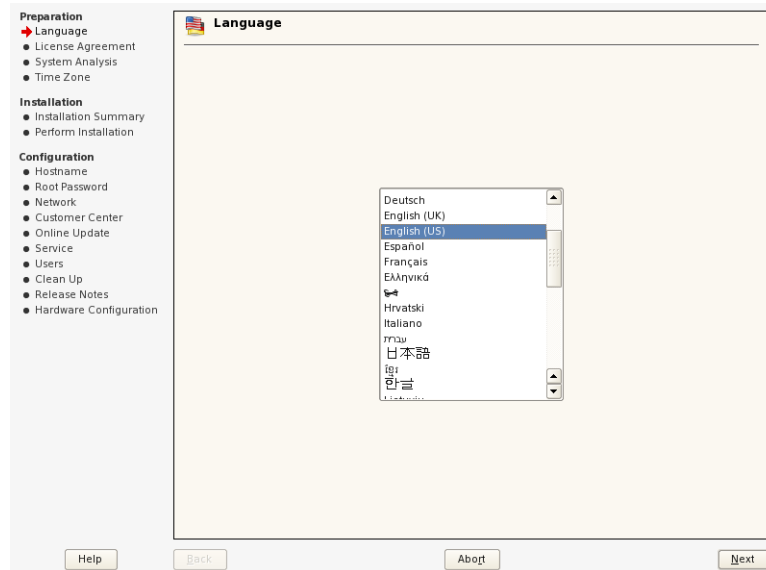
Select the **Installation** option to start the installation process. If the installation fails for some reason, try to install with the options **Installation - ACPI Disabled**, **Installation - Local APIC Disabled**, or **Installation - Safe Settings**.

After you select an installation option, a minimal Linux system loads to run the YaST installation program.

Select the System Language

After YaST starts, the following appears:

Figure 1-2



Almost all YaST installation dialogs use the same format:

- The left side displays an overview of the installation status.
- From the lower left side, you can select a help button to get information about the current installation step.
- The right side displays the current installation step.
- The lower right side provides buttons for navigating to the previous or next installation steps, or for aborting the installation.



If the installation program does not detect your mouse, you can use the Tab key to navigate through the dialog elements, the arrow keys to scroll in lists and Enter to select buttons. You can change the mouse settings later in the installation process.

From the language dialog, select the language of your choice, and then select **Next** to continue to the next step, the License Agreement.

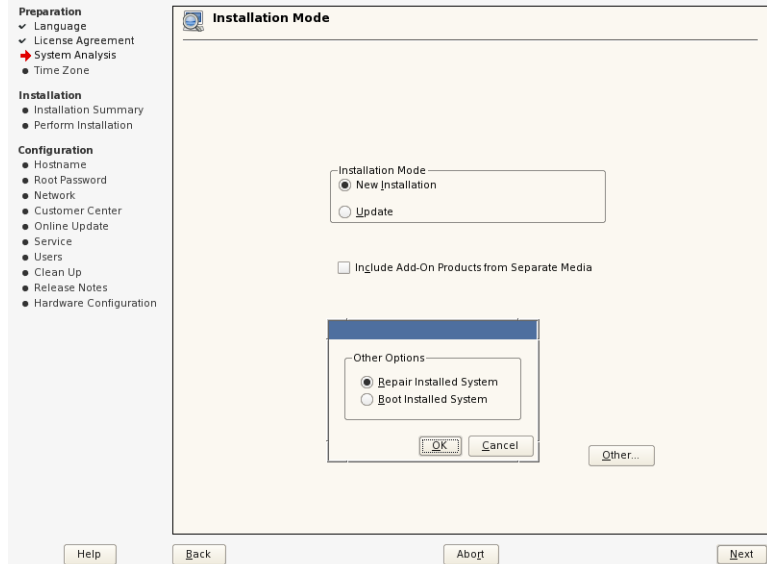
You have to select **Yes, I Agree to the License Agreement** to get to the next step by selecting **Next**.

Select the Installation Mode

If there is no operating system installed on your computer, the installation mode dialog offers only **New Installation**. (Update and Other cannot be selected in this case.)

If YaST detects another SUSE Linux installation, you are offered more options, some of which are only available after selecting **Other**, like in the following:

Figure 1-3



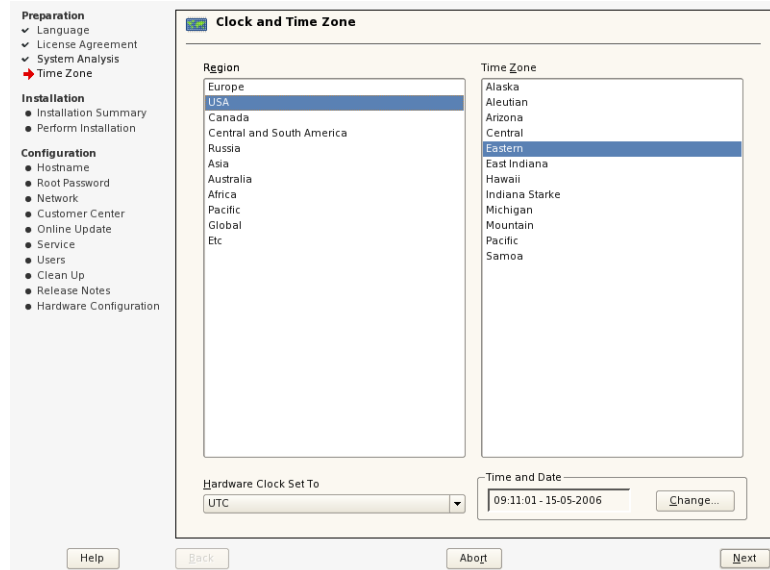
- **New installation.** Performs a normal new installation of SLES 10. This is the default option.
- **Update.** Updates a previously installed SLES 9 installation.
- **Other.** Offers two more options:
 - **Repair Installed System.** Repairs a previously installed SLES 10 installation.
 - **Boot Installed System.** Boots a previously installed Linux installation.
- **Abort Installation.** Terminates the installation process.

For a normal installation, select **New Installation** and then select **Next** to proceed to the next step.

Set the Clock and Time Zone

YaST selects the time zone of the installed system according to your language selection. Change the time zone if you are located in a different one.

Figure 1-4

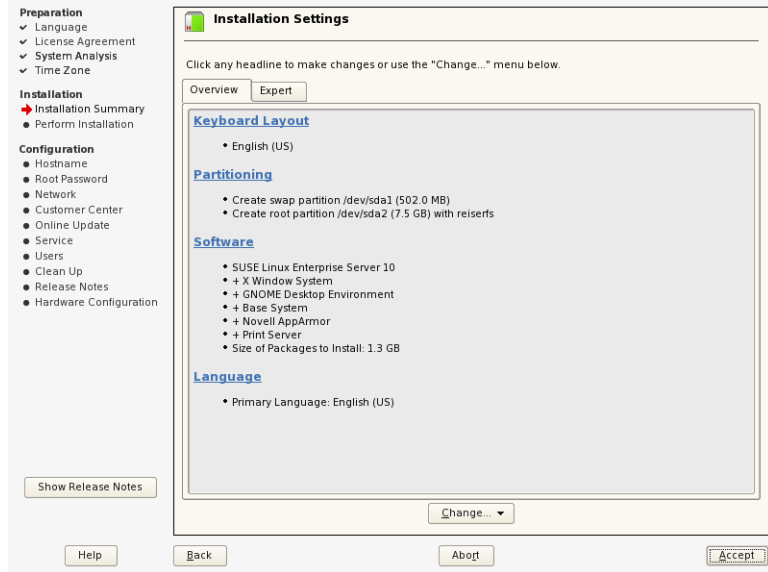


If your hardware clock is set to UTC (Universal Time Coordinated) the system time is set according to your time zone and automatically adjusted to daylight saving time. If your hardware clock is set to local time, select Local Time instead of UTC in the drop-down menu.

Understand and Change the Installation Settings

YaST analyzes the system and creates an installation proposal. The proposed settings are displayed on two tabs, as in the following figure; **Overview** shows the main categories:

Figure 1-5



The proposal displays installation settings that are necessary for a base installation. You can change these settings by selecting the following headings:

- **Keyboard layout.** Changes the keyboard layout. YaST selects the keyboard layout according to your language settings. Change the keyboard settings if you prefer a different layout.
- **Partitioning.** Changes the hard drive partitioning. If the automatically generated partitioning scheme does not fit your needs, you can change it by selecting this headline.
- **Software.** Changes the software selection. You can select or deselect software.

- **Language.** Changes the default language.

The Experts tab shows the above options, plus the following:

- **System.** Restarts the hardware detection process and displays a list of all available hardware components. You can change the PCI-ID setup, select single components and view details, or save the list to a file.
- **Add-on Products.** Choose this option to include any add-on products.
- **Booting.** Select this if you want to change any boot loader settings or use Lilo (Linux Loader) instead of GRUB (Grand Unified Bootloader) as boot loader.
- **Time zone.** Opens the Clock and Time Zone dialog described earlier.
- **Default Runlevel.** Changes the runlevel. If a graphical environment is installed, the default is runlevel 5, otherwise it is 3.

Of the settings described above, partitioning and software will be discussed in more detail.

Verify Partitioning

In most cases, YaST proposes a reasonable partitioning scheme that you can accept without change. However, you might need to change the partitioning manually if

- You want to optimize the partitioning scheme for a special purpose server (such as a file server).
- You want to configure LVM (Logical Volume Manager).
- You have more than one hard drive and want to configure RAID (Redundant Array of Independent Disks).

- You want to delete existing operating systems so you have more space available for your SLES 10 installation.

To partition the hard drive manually, you need to know the following:

- The Basics of Hard Drive Partitioning
- The Basic Linux Partitioning Scheme
- Change YaST's Partitioning Proposal
- Use the YaST Expert Partitioner

The Basics of Hard Drive Partitioning

Partitions divide the available space of a hard drive into smaller portions. This lets you install more than one operating system on a hard drive or use different areas for programs and data.

Every hard disk (on an Intel platform) has a partition table with space for four entries. An entry in the partition table can correspond to a primary partition or an extended partition. However, only one extended partition entry is allowed.

A *primary partition* consists of a continuous range of cylinders (physical disk areas) assigned to a particular file system. If you use only primary partitions, you are limited to four partitions per hard disk (because the partition table can only hold four primary partitions).

This is why extended partitions are used. *Extended partitions* are also continuous ranges of disk cylinders, but can be subdivided into logical partitions. *Logical partitions* do not require entries in the main partition table. In other words, an extended partition is a container for logical partitions.

If you need more than four partitions, create an extended partition instead of a fourth primary partition. This extended partition should include the entire remaining free cylinder range. Then create multiple logical partitions within the extended partition. The maximum number of partitions is 15 on SCSI disks and 63 on (E)IDE disks.

It does not matter which type of partitions you use on Linux systems; primary and logical partitions both work well.

The Basic Linux Partitioning Scheme

The optimal partitioning scheme for a server depends on the purpose of the server.

A SLES 10 installation needs at least two partitions:

- **Swap partition.** This partition is used by Linux to move unused data from the main memory to the hard drive, thus freeing main memory which then can be used by other processes.
- **Root partition.** This is the partition that holds the top (/) of the file system hierarchy, the so-called root directory.

No matter what partition scheme you choose, you always need at least one swap partition and a root partition.

The following guidelines help you determine what you can install depending on the space available on your hard disk for your file system:

- **800 MB.** This allows for a minimal installation with no graphical interface. With this configuration, you can only use console applications.
- **1300 MB.** This allows for an installation with a minimum graphical interface. This includes the X Window system and a few graphical applications.

- **2 GB.** This holds the default installation proposed by YaST. This configuration includes a modern desktop environment (such as KDE or GNOME), and provides enough space for several additional applications.
- **4 GB.** This allows for a full installation, including all software packages shipped with SLES 10.

You can put certain directories on separate partitions. If you do this, your root partition can be smaller than outlined above. Any space for data needs to be added to the above.

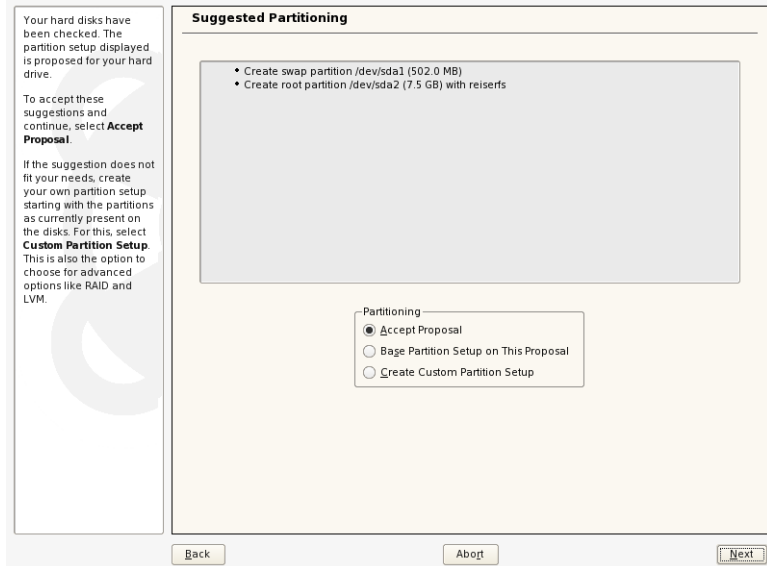


As today's computers are equipped with hard disks with capacities of 100 GB and more, there is still plenty of space for data. Considering the difficulties involved with changing partitions in an installed system and the size of current hard disks, you should therefore allocate much more space than the above minimum when deciding on the hard disk layout.

Change YaST's Partitioning Proposal

To use YaST to change the partition scheme, select the **Partitioning** headline in the installation proposal. The following appears:

Figure 1-6

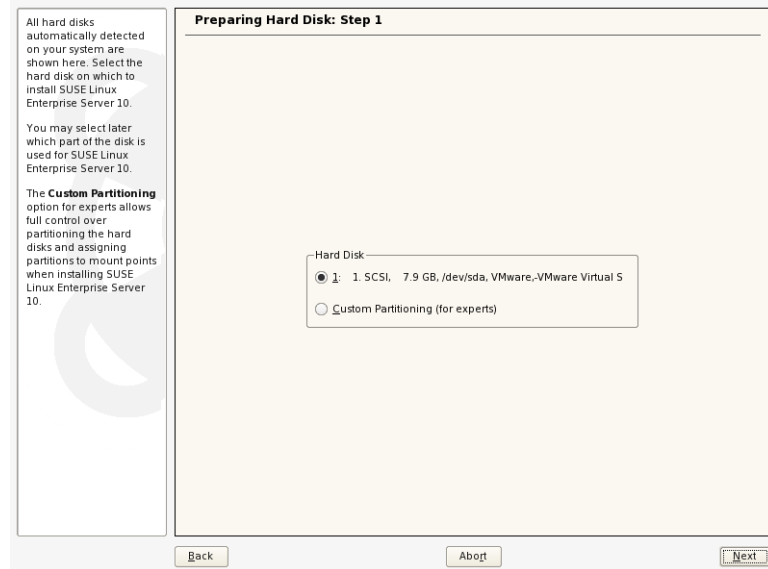


In the top part of the dialog, YaST displays the automatically generated partitioning proposal. The lower part of the dialog provides the following options:

- **Accept Proposal.** Accepts the partitioning scheme and returns to the main installation proposal.
- **Base Partition Setup on This Proposal.** Starts the YaST Expert Partitioner, using the partition proposal as base setup.

■ **Create Custom Partition Setup.** Displays the following:

Figure 1-7



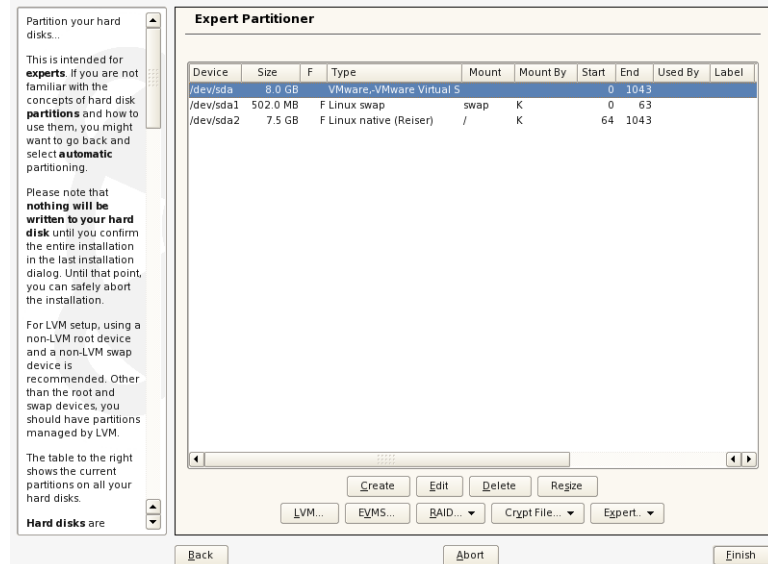
In this dialog, you can select

- ❑ A hard disk; selecting Next opens a dialog where you can choose to use the entire hard disk or some of the existing partitions for the installation of SLES 10.
- ❑ **Custom Partitioning**; selecting Next opens the YaST Expert Partitioner, displaying the existing partition layout.

Use the YaST Expert Partitioner

When you start the YaST Expert Partitioner, the following appears:

Figure 1-8



In the top part of the dialog, YaST lists details of the current partition setup. Depending on your previous choice, the list contains the partitioning proposal created by YaST or the current physical disk setup.

The buttons in the lower part of the dialog are used to create, edit, delete, and resize partitions, as well as to administer LVM (Logical Volume Manager), EVMS (Enterprise Volume Management System), RAID (Redundant Array of Independent Disks).



The changes made with the YaST Expert Partitioner are not written to disk until the installation process is started. You can always discard your changes by selecting **Back** or you can restart the Expert Partitioner to make more changes.

The following entries are displayed for every hard disk in your system:

- One entry for the hard disk itself, which has the corresponding device name in the Device column (such as **/dev/sda**).
- One entry for every partition on the hard disk with the corresponding device name and the partition number in the Device column (such as **/dev/sda1**).

Each entry in the list includes information in the following columns:

- **Device.** Displays the device name of the hard disk or the partition.
- **Size.** Displays the size for the hard disk or partition.
- **F.** When the character “F” is displayed in this column, the partition will be formatted during the installation process.
- **Type.** Displays the partition or hard disk type. Depending on the operating system and the architecture, partitions can have various types, like Linux native, Linux swap, Win95 FAT 32, NTFS, etc.
- **Mount.** Displays the mount point of a partition. For swap partitions, the keyword *swap* is used instead.
- **Mount By.** Indicates how the file system is mounted: K—Kernel Name, L—Label, U—UUID, I—Device ID, and P—Device Path.
- **Start.** Displays the start cylinder of a hard disk or partition. Hard disk entries always start with 0.
- **End.** Displays the end cylinder of a hard disk or partition.
- **Used By.** This column holds information about the system using this partition, like LVM-system.
- **Label, Device ID, Device Path.** These columns list the respective information.

The buttons in the lower part of the dialog let you

- Create New Partitions
- Edit Existing Partitions
- Delete Existing Partitions
- Resize Existing Partitions
- Perform Expert Tasks

These administrative tasks are covered in more detail below. EVMS (<http://evms.sourceforge.net/>) and Crypt File Partitions are not covered in this course.

Create New Partitions

Create a new partition by selecting **Create**. A dialog with one of the following options appears (the options you see depend on your hard disk setup):

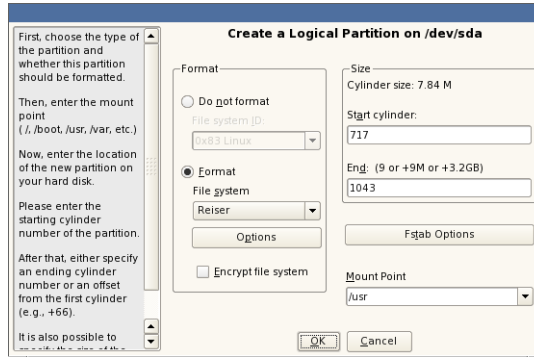
- If you have more than one disk in your system, you are asked to select a disk for the new partition first.
- If you do not have an extended partition, you are asked if you want to create a primary or an extended partition.
- If you have an extended partition, and there is space on the hard drive outside the extended partition for additional primary partitions, you are asked if you want to create a primary or a logical partition.
- If you have 3 primary partitions and an extended partition, you can only create logical partitions.



You need enough space on your hard disk to create a new partition. You learn later in this section how to delete existing partitions to free used disk space.

If you choose to create a primary or a logical partition, the following appears:

Figure 1-9



This dialog provides the following options:

- **Format.** This lets you choose one of the following options:
 - **Do not format.** Do not format the newly created partition. No file system will be created on this new partition. You can select the partition type in the drop-down list.
 - **Format.** Formats the new partition with the file system you select from the File System drop-down list.

You can choose from the following file systems:

- **Ext2.** Formats the partition with the Ext2 file system. Ext2 is an old and proven file system, but it does not include journaling.
- **Ext3.** Formats the partition with the Ext3 file system. Ext3 is the successor of Ext2 and offers a journaling feature.
- **Reiser.** Formats the partition with ReiserFS, a modern journaling file system. (This is the default option.)

- ❑ **FAT.** Formats the partition with the FAT file system. FAT is an older file system used in DOS and Windows. You can use this option to create a data partition which is accessible from Windows and Linux. You must not create a root partition with this file system.
- ❑ **XFS.** Formats the partition with XFS, a journaling file system originally developed by SGI.
- ❑ **Swap.** Formats the partition as a swap partition.

If you are not sure which file system to choose, select Reiser for root and data partitions and Swap for swap partitions.

- ❑ **Options.** By selecting Options, you can change parameters for the file system you selected. You can use the default parameters in most cases.
 - ❑ **Encrypt file system.** If you select this option, the partition with the file system is encrypted. Encrypting a file system prevents unauthorized mounting only; once mounted the files are accessible like any other file on the system.

You should only use this option for non-system partitions such as user home directories.
- **Size.** Lets you configure the size of the new partition with the following:
 - ❑ **Start Cylinder.** Determines the first cylinder of the new partition. YaST normally preselects the first available free cylinder of the hard disk.
 - ❑ **End.** Determines the size of the new partition. YaST normally preselects the last available free cylinder.

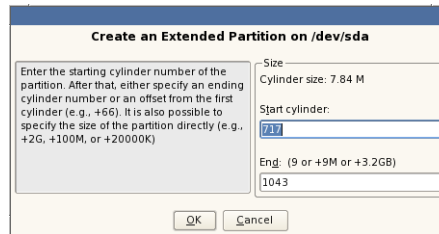
To configure the end cylinder, do one of the following:
 - ❑ Enter the cylinder number.

- ❑ Enter a plus sign (+) followed by the amount of disk space for the new partition. Use M for MB and G for GB. YaST calculates the last cylinder number. For example, enter **+5G** for a partition size of 5 GB.
- **Fstab Options.** Select this option to edit the fstab entry for this partition. The default setting should work in most cases.
- **Mount Point.** Select the mount point of the new partition from this drop-down list. You can also enter a mount point manually, if it's not available in the list. The mount point will be created automatically during installation.

After changing the parameters, select **OK** to add the new partition to the partition list.

If you chose to create an extended partition, the following appears:

Figure 1-10



You can enter the following:

- **Start cylinder.** The start cylinder determines the first cylinder of the new partition. YaST normally preselects the first available free cylinder of the hard disk.
- **End.** The end cylinder determines the size of the new partition. YaST normally preselects the last available cylinder of the hard disk.

To configure the end cylinder, do one of the following:

- ❑ Enter the cylinder number.

- Enter a plus sign (+) followed by the amount of disk space for the new partition. Use M for MB and G for GB. YaST calculates the last cylinder number.

For example, enter **+5G** for a partition size of 5 GB.

After entering the size, select **OK** to add the new extended partition to the partition list.

Edit Existing Partitions

Select a partition from the list and select **Edit**. You can edit only primary and logical partitions with the Expert Partitioner. You cannot edit extended partitions or the entry for the entire hard disk.

If you edit a primary or logical partition, a dialog appears which is very similar to the Create Partition dialog described above. You can change all options except the partition size.

After changing the partition parameters, select **OK** to save your changes to the partition list.

Delete Existing Partitions

To delete a partition, select a partition from the list, select **Delete**, and then select **Yes** in the confirmation dialog. The partition is deleted from the partition list.

Remember that you also delete all logical partitions when you delete an extended partition.

If you select the entry for the entire hard disk and select **Delete**, all partitions on the disk are deleted.

Resize Existing Partitions

Select a partition from the list and select **Resize**.



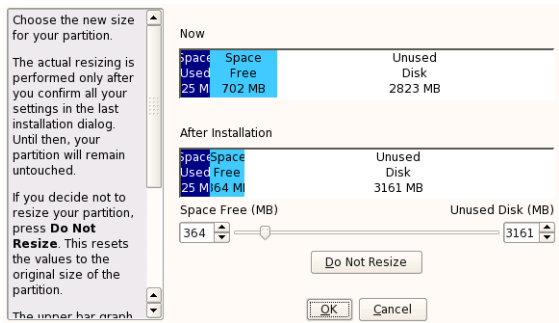
Although you can reduce a partition's size without deleting it to increase free space on the hard disk, you should always back up the data on the partition before resizing it.



If the selected partitions are formatted with the FAT or NTFS file system, there are certain steps you should take in Windows before resizing (scandisk and defrag). See the section on installation in the *SUSE Linux Enterprise Server 10 Administration Manual* (/usr/share/doc/manual/sles-admin_en/, package sles-admin_en) for details.

After you select **Resize**, the following appears:

Figure 1-11



This dialog includes the following:

- Two bars representing the partition before and after the resizing process
 - **Now.** Used space is designated by dark blue and the available space by light blue. If there is space not assigned to a partition it is designated by white.

- ❑ **After installation.** Used space is designated by dark blue and the free space by light blue. The space that is available for a new partition is designated by white.
- A slider to change the size of the partition
- Two text fields that display the amount of free space on the partition being resized and the space available for a new partition after the resizing process
- A Do Not Resize button used to reset the partition to the original size

To resize the partition, move the slider until enough unused disk space is available for a new partition. When you select **OK**, the partition size changes in the partition list.

Perform Expert Tasks

When you select **Expert**, the following options are available:

- **Reread the Partition Table.** Resets the partition list to the current physical disk setup. All changes will be lost.
- **Import Mount Points from Existing /etc/fstab.** Scans the hard disks for an /etc/fstab file. You can load this file and set the mount points accordingly.
- **Delete Partition Table and Disk Label.** Deletes the partition table and the disk label of the selected hard disk. *All data on that disk will be lost.*

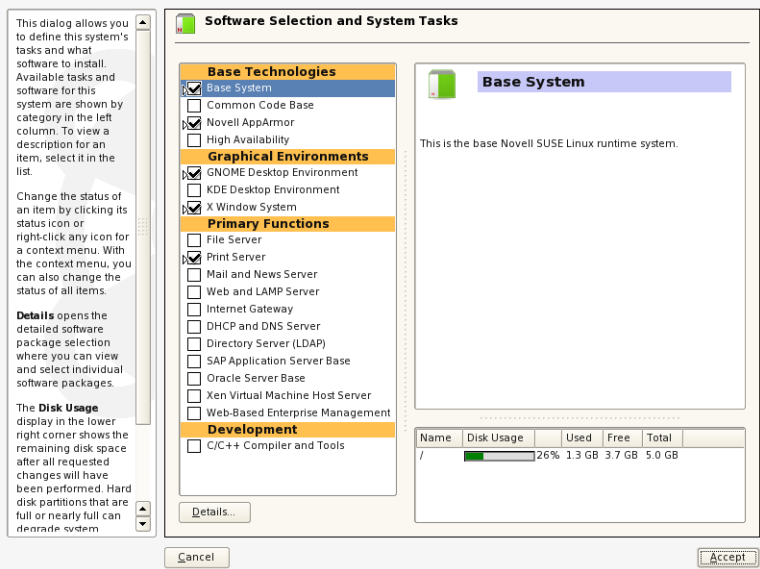
When you finish configuring settings in the Expert Partitioner, return to the installation proposal by selecting **Finish**.

Select Software

SLES 10 contains many software packages for various application purposes. Instead of selecting needed packages one by one, you can select various software categories.

Depending on the available disk space, YaST preselects several of these categories. Selecting **Software** in the installation overview opens the following dialog:

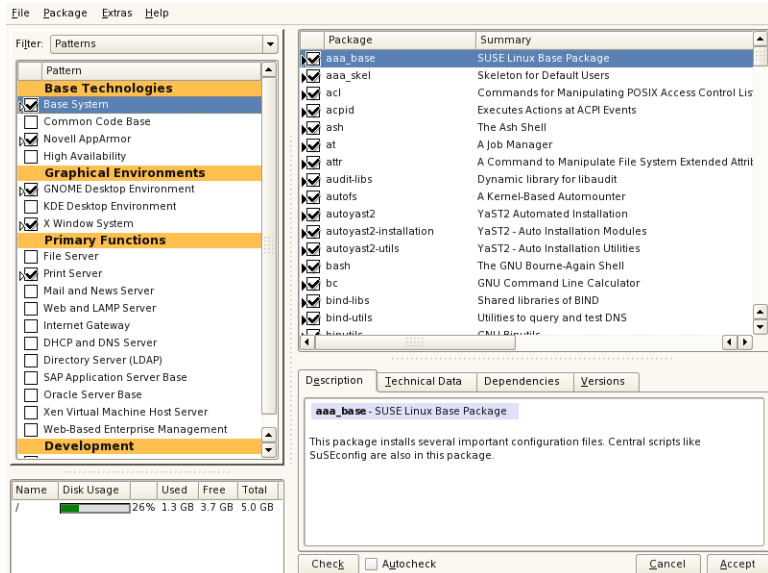
Figure 1-12



The figure above shows the default selection. A brief description appears on the right when you highlight a category in the center column.

To find out which packages are contained in the various categories, select Details, which opens the following dialog:

Figure 1-13



Selecting one pattern on the left shows the software packages contained in that category on the right. Selecting the square to the left of the pattern selects it for installation or deselects it.

A package typically contains an application and all additional files required to use the software. Sometimes larger applications can be split into multiple packages and several small applications can be bundled into a single package. SUSE Linux Enterprise Software uses the RPM Package Manager for software management.

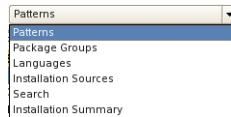
Sometimes one software package needs another one to run. Information on these dependencies is stored in the RPM packages. YaST can automatically select software packages when another package requires them.

You can install a package by selecting the check box for that package in the package list on the right.

To view details for a package, highlight its entry in the package list. The details for the currently selected package are displayed below the package list.

The Filter drop-down menu offers different views on the software packages available and the software scheduled for installation.

Figure 1-14



- **Patterns.** This leads to the dialog shown in Figure 1-13.
- **Package Groups.** Displays the packages in a hierarchical tree view. There are main categories, like Productivity, Programming, System, Hardware, etc. and subcategories. Selecting a category on the left displays the software packages belonging to that category on the right.
- **Languages.** You can select support for additional languages.
- **Installation Sources.** Displays the installation sources configured.
- **Search.** Displays a search dialog to search for packages.
- **Installation Summary.** Displays a summary of the packages selected for installation.

The disk usage of the software packages selected for installation is displayed in the lower left corner of the dialog.

Select the option **Check** to check the dependencies of the selected packages. This check is also done when you confirm the package selection dialog.

If the check box **Autocheck** is selected, dependencies are checked every time you select or deselect a package.

Confirm your package selection and return to the installation proposal by selecting **Accept**.

Start the Installation Process

After customizing the installation proposal, select **Accept**. A dialog appears asking you to confirm the proposal. Start the installation process by selecting **Install**; return to the installation proposal by selecting **Back**.

Before installing software packages, YaST changes the hard disk partitioning.

Depending on your software selection and the performance of your system, the installation process takes 15–45 minutes.

If you are using the product CD set instead of the DVD, YaST asks you to change the installation CDs. Insert the requested CD and continue the installation by selecting **OK**.

After all software packages are installed, YaST reboots the computer and prompts you for the host name, root password, network configuration details, etc., to further customize your installation.

Objective 2 **Configure the SLES 10 Installation**

In this part of the installation process, you use YaST to perform the following configuration tasks:

- Set the Host Name
- Set the root Password
- Configure the Network
- Test the Internet Connection
- Novell Customer Center Configuration and Online Update
- Manage Users
- Configure Network Services
- Configure Hardware
- Finalize the Installation Process

Set the Host Name

YaST suggests a host name **linux-xxxx**, with **xxxx** being composed of random characters. The domain defaults to **site**. Change the host and the domain name to the correct values for the computer and remove the check mark in front of **Change Hostname via DHCP**.

If the computer gets its host name and domain via DHCP you do not need to change anything in this dialog.

Set the root Password

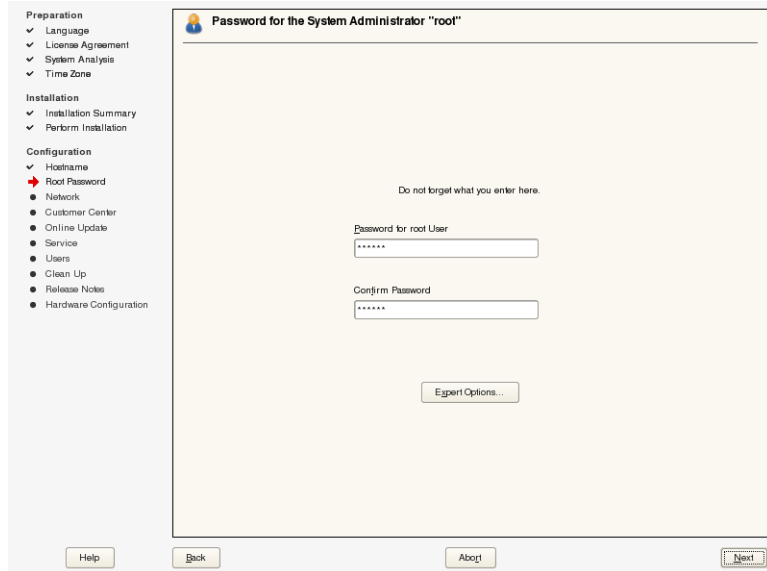
root is the name of the administrator of the system. Unlike regular users, who might not have permission to do certain things on the system, *root* has unlimited power to do anything, including the following:

- Access every file and device in the system.
- Change the system configuration.
- Install software.
- Set up hardware.

The *root* account should only be used for system administration, maintenance, and repair. Logging in as *root* for daily work is risky: a single mistake can lead to irretrievable loss of many system files.

To let you set the root password during the installation process, YaST displays the following:

Figure 1-15



Enter the same password in both text fields of the dialog.

You should choose a password that cannot be guessed easily. Use numbers, lowercase and uppercase characters to avoid dictionary attacks.

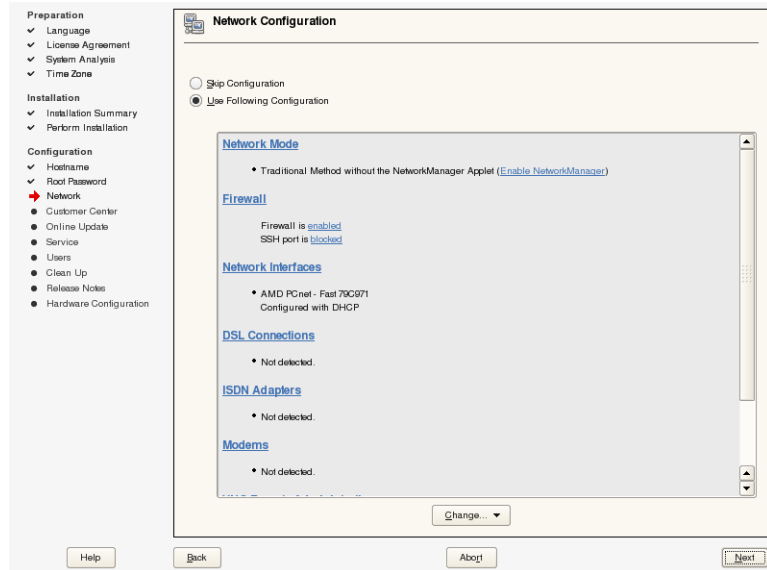
By selecting Expert Options, you can choose the password encryption algorithm. In most cases you can use the default setting, which is **Blowfish**.

After entering the root password, continue to the next configuration step by selecting **Next**. In case your password is too simple or weak, you are shown a warning. Go back to enter a better password, or accept the weakness and go on.

Configure the Network

To let you configure the network connection of your system, YaST displays the following:

Figure 1-16



In the top part of the dialog, you can choose one of the following options:

- **Skip Configuration.** Skip the network configuration for now. You can configure the network connection later in the installed system.
- **Use Following Configuration.** Use the network configuration proposal displayed in the area below.

The network configuration proposal is similar to the installation proposal at the beginning of the base installation, with headings that can be selected to view and configure further details. and includes the following entries:

- **Network Mode.** Switch between the traditional method of managing the network and network manager. On a server use the traditional method. Network manager is more suitable for a notebook, enabling users to easily switch between for instance Ethernet and wireless access.
- **Firewall.** If you want to administer the computer via SSH, toggle **SSH port is blocked** to **SSH port is open** by clicking on **blocked**. You can disable the firewall by clicking on **enabled**. When the firewall is disabled, then SSH is accessible as well.

Selecting **Firewall** itself opens a dialog allowing detailed configuration of the firewall settings.

- **Network Interfaces.** Displays the network interfaces found (such as Ethernet or a Wireless-LAN adapter) and their configuration (like DHCP).
- **DSL Connections.** Displays the configuration of DSL devices. These can be DSL modems connected with an Ethernet adapter or internal DSL modems.
- **ISDN Adapters.** Displays the configuration of ISDN devices.
- **Modems.** Displays the configuration of analog modems.
- **VNC Remote Administration.** Displays the configuration of remote administration using VNC.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

- **Proxy.** Displays the HTTP and FTP proxy settings.

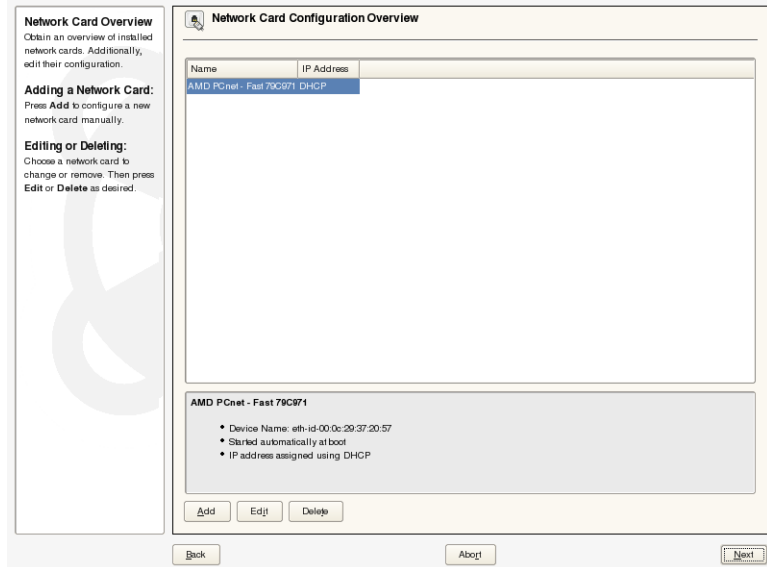
You can change a configuration by selecting the headline of the entry or by selecting the entry from the **Change** drop-down list. This menu also lets you reset all settings to the defaults generated by YaST.

If you are not sure which settings to use, stay with the defaults generated by YaST.

Configure Network Interfaces

After starting the network interface configuration, YaST displays the **Network Card Configuration Overview**. It lists all network cards, the configured ones as well as those which are not yet configured:

Figure 1-17



The upper part lists the cards found, the lower part show details for the network card that is highlighted in the upper part.

At this point, you can do one of the following:

- Add a Network Card Manually
- Edit an Existing Configuration
- Delete Existing Partitions

Add a Network Card Manually

If you want to configure a network card that was not automatically detected, select **Add** to display the following:

Figure 1-18

Here, set up your networking device. The values are written to `/etc/sysconfig/hardware/hw`.

Options for the module should be written in the format `option=value`. Each entry should be space separated, for example, `ip=0x300 irq=5`.

Note: If two cards are configured with the same module name, options will be merged while saving.

Get a list of available network cards by pressing **Select from List**.

If you have a **PCMCIA** network card, select PCMCIA. If you have a **USB** network card, select USB.

Manual Network Card Configuration

Network Configuration

Device Type: Ethernet Configuration Name: 0

Kernel Module

Hardware Configuration Name: ethic0

Module Name: Options:

☐ PCMCIA ☐ USB

Select from List

Back Abort Next

From this dialog, you can configure the following:

- **Network Configuration.** Specify the network **Device Type** (Ethernet, Bluetooth, Wireless, etc.) and the device number.

- **Kernel Module.** If your network card is a PCMCIA or USB device, select the corresponding check boxes and confirm selecting **Next**.
- Otherwise, select **Select from List** and select your network card from the list. YaST automatically loads the appropriate driver for the selected card. Confirm by selecting **OK**.
- If you selected **Wireless** as Device Type for a WLAN card, **Next** brings you to a Network Address dialog. The default, DHCP, is usually the right choice. Selecting **Next** again opens a dialog where you can enter WLAN specific configuration parameters, like the **Operating Mode**, the **Network Name** (ESSID), the **Authentication Mode**, and the encryption key.

WEP keys are entered in a separate dialog after selecting WEP Keys. Expert settings concern parameters like the bit rate.

When you are finished with this dialog, select **Next**, which returns you to the Network Card Configuration Overview.

Edit an Existing Configuration

To edit a network card configuration, highlight its entry in the upper part of the **Network Card Configuration Overview** and select **Edit**.

Figure 1-19

You can select dynamic address assignment if you have a **DHCP server** running on your local network.

Also select this if you do not have a static IP address assigned by the system administrator or your cable or DSL provider.

Network addresses are then obtained **automatically** from the server.

Otherwise, network addresses must be assigned **manually**.

Enter the IP address (e.g., 192.168.100.99) for your computer, the network mask (usually 255.255.255.0), and, optionally, the default gateway IP address.

Clicking **Next** completes the configuration.

Contact your **network administrator** for more information about the network configuration.

Network Address Setup

General Address

Device Type: Ethernet Configuration Name: id-00-0c:29-37:20:57

☒ Automatic Address Setup (via DHCP)
☐ Static Address Setup

IP Address:

Subnet Mask:

Detailed Settings

Hostname and Name Server:

Routing:

Advanced...:

Back Abort Next

The **Address** tab offers the following configuration options:

- **Automatic Address Setup (via DHCP).** If your network has a DHCP server, you can set up your network address automatically. You should also use this option if you are using a DSL line with no static IP address assigned by the ISP.

If you decide to use DHCP, you can configure the details after selecting **DHCP Options** from the **Advanced** drop-down list. Specify whether the DHCP server should always broadcast its responses (in this case select **Request Broadcast Response**) and any identifier to use.

By default, DHCP servers use the network card's hardware address to identify an interface. If you have a virtual host setup where different hosts communicate through the same interface, an identifier is necessary to distinguish them.

- **Static Address Setup.** If you have a static address, select the corresponding check box. Then enter the address and subnet mask for your network. The preset subnet mask should match the requirements of a typical home network.
- **Hostname and Name Server.** Select this option to set the host name and the name server manually.
- **Routing.** Select this option to configure routing manually.

The **General** tab offers the following configuration options:

- **Firewall Zone.** Decide whether this interface belongs to the Internal, External, or Demilitarized Zone, or if all traffic should be blocked (No Zone).
- **Device Activation.** Choose from At Boot Time, On Cable Connection, On Hotplug, Manually, or Never.
- **Detailed Network Interface Settings.** Specify the Maximum Transfer Unit (MTU), which sometimes improves the performance of certain DSL (Digital Subscriber Line) connections. For PPPoE (Point-to-Point over Ethernet) values between 1400 and 1492 are common; these values vary, depending on your ISP (Internet Service Provider).

Confirm the Network Address Setup and return to the Network Card Configuration Overview by selecting **Next**.

Delete an Existing Configuration

To delete an existing configuration, highlight it in the upper part of the Network Card Configuration Overview and select **Delete**.

When finished with adding, editing, or deleting network card configurations, save the network device configuration and return to the Network Configuration proposal by selecting **Next**.

After finishing the Network Configuration, select **Next**.

Test the Internet Connection

YaST then asks you to test your connection to the Internet. Select one of the following options:

- **Yes, Test Connection to the Internet.** YaST tries to test the Internet connection by downloading the latest release notes and checking for available updates.
If you select this option, the results are displayed on the next dialog.
- **No, Skip This Test.** Skip the connection test. If you skip the test, you cannot update the system during installation.

Select one of the options and select **Next**.

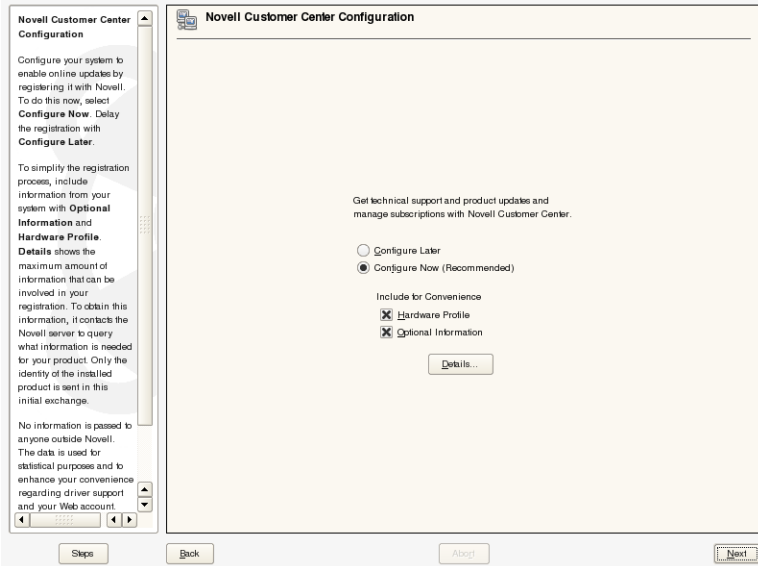
If the test fails, you can view the logs to find out what went wrong.

Select **Next** to continue.

Novell Customer Center Configuration and Online Update

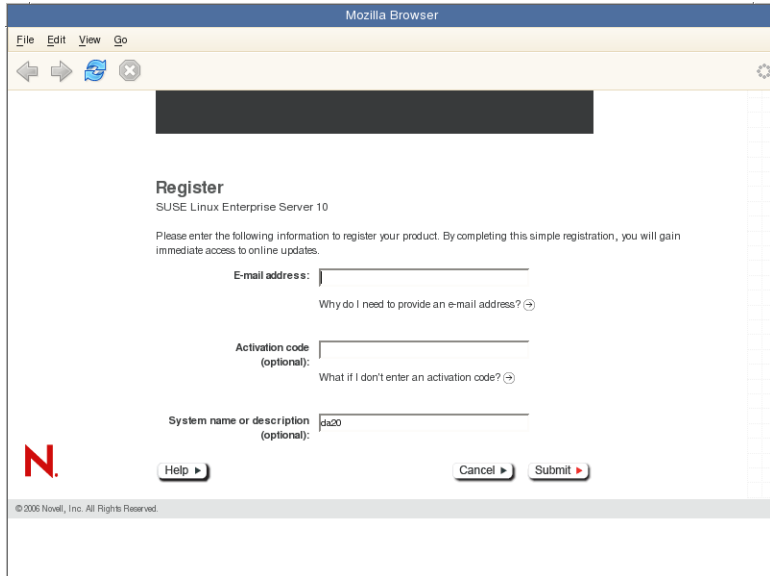
If the Internet connection test was successful, you can configure the Novell Customer Center, which is required to perform an online update. If there are any update packages available on the SUSE update servers, you can download and install them to fix known bugs or security issues.

Figure 1-20



Selecting **Next** starts a Browser and connects to the Novell web site, where all you have to enter is your email address, and an activation code, if available.

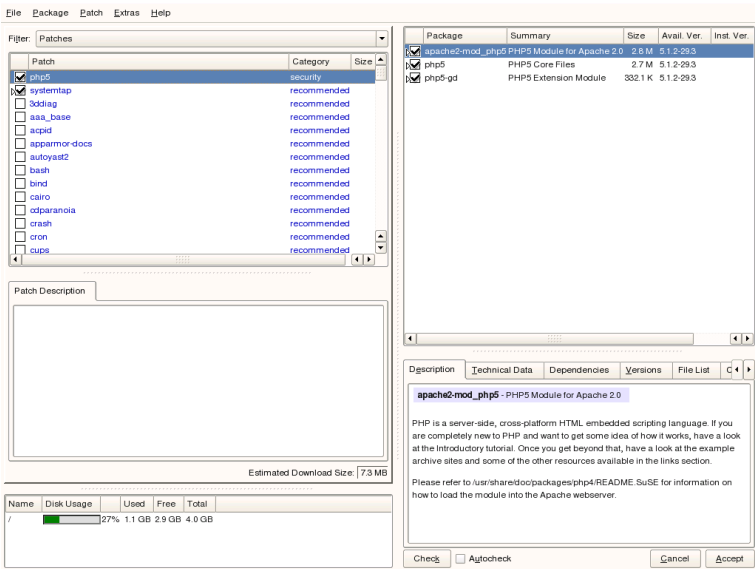
Figure 1-21

The image is a screenshot of a Mozilla Browser window. The title bar says "Mozilla Browser". The address bar is empty. The main content area displays a registration form for "SUSE Linux Enterprise Server 10". The form has a heading "Register" and a subheading "SUSE Linux Enterprise Server 10". Below this, it says "Please enter the following information to register your product. By completing this simple registration, you will gain immediate access to online updates." The form contains three input fields: "E-mail address:", "Activation code (optional):", and "System name or description (optional):". The "System name or description" field has the value "ja20" entered. To the left of the form is a large red "N" logo. At the bottom of the form are three buttons: "Help", "Cancel", and "Submit". Below the form, there is a small copyright notice: "© 2006 Novell, Inc. All Rights Reserved." The browser's status bar at the bottom is empty.

After successful registration, the Online Update dialog opens. You can start the Online Update by selecting **Run Update** and **Next**. (You can also select **Skip Update** to perform the update later in the installed system.)

YaST's online update dialog opens up with a list of available patches (if any).

Figure 1-22



Select the patches you want to install, and then start the update process by selecting **Accept**.

Once the installation is complete, visit the Novell Customer Center at <http://www.novell.com/center/> to administer your Novell products and subscriptions.

Configure Network Services

In the next installation step, YaST displays the Service Installation Settings dialog.

In the top part of the dialog, you can choose one of the following options:

- **Skip Configuration.** Skip this configuration step. You can enable the services later in the installed system.
- **Use Following Configuration.** Use the automatically generated configuration displayed below this option or select one of the following headlines to change the configuration:

- **CA Management.** The purpose of a CA (certification authority or certificate authority) is to guarantee a trust relationship among all network services that communicate with each other.

If you decide that you do not want to establish a local CA, you must secure server communications using SSL (Secure Sockets Layer) and TLS (Transport Layer Security) with certificates from another CA.

By default, a CA is created and enabled during the installation.

To create proper certificates, the hostname has to be set correctly earlier in the Network Interface Configuration; otherwise the generated certificate will contain an incorrect hostname.

- **LDAP Server.** You can run an LDAP (Lightweight Directory Access Protocol) server on your host to have a central service managing a range of configuration settings. Typically, an LDAP server handles user account data, but since SLES 9 you can also use LDAP for mail, DHCP, and DNS related data on SUSE Linux Enterprise Server.

By default, an LDAP server is not set up during installation.

If you are not sure about the correct settings, keep the defaults generated by YaST. You can change the configuration later in the installed system.

When you are finished, select **Next**.

Manage Users

To manage users during this configuration step, do the following:

- Select the Authentication Method
- Configure the Authentication Method

Select the Authentication Method

The Authentication Method dialog offers four methods: You can selecting one of the following options:

- **Local (/etc/passwd).** Select this option to configure the system to use the traditional file-based authentication method.
- **LDAP.** If you have an LDAP server in your network, you can configure your system as an LDAP client.
- **NIS.** If you have a NIS server in your network, you can configure your system as a NIS client.
- **Windows Domain.** Choose this if you want to authenticate against a Windows Server.

If you are not sure which method to select, stay with **Local**, which is the default for SLES 10.

After selecting an authentication method, select **Next**.

Configure the Authentication Method

The next dialog differs, depending on which authentication method you selected. We will cover here:

- Add Local Users
- Configure the System as an LDAP Client

The dialogs for NIS and Windows Domain are used in a similar fashion to obtain the necessary information to enable the respective authentication method.

Add Local Users

If you select **Local** as the authentication method, the following appears:

Figure 1-23

Enter the User's Full Name, Username, and Password to assign to this user account.

When entering a password, distinguish between uppercase and lowercase. Passwords should not contain any special characters, such as accented characters.

With the current password encryption (Blowfish), the password length should be between 5 and 72 characters.

Valid password characters are letters, digits, blanks, and `!@#$%^&*~.-_+=`~'"/>{[]}~`.

To ensure that the password was entered correctly, repeat it exactly in a second field. Do not forget your password.

Create the **User Login** from components of the full name by clicking **Suggestion**. It may be modified, but use only letters (no accented characters), digits, and `~.-_`. Do not use `uppercase letters` in this

New Local User

User's Full Name
[Text Field]

Username
[Text Field] **Suggestion**

Password
[Text Field]

Confirm Password
[Text Field]

☐ Receive System Mail
☐ Automatic Login

User Management

Skip **Back** **Next** **Apply**

You can use the following in this dialog to add local users to the system (account information is stored in the files `/etc/passwd` and `/etc/shadow`):

- **User Data.** Enter the full user name, the login name, and the password.

To provide effective security, a password should be 8 or more characters long. The maximum length for a password ranges from 8 to 128 characters, depending on the algorithm used to hash the password. While the Crypt algorithm commonly used in the past used only the first eight characters of the password, more recent algorithms allow longer passwords.

Passwords are case-sensitive. Special characters are allowed, but they might be hard to enter depending on the keyboard layout.

- **Password Settings.** Select this option to change advanced password settings (such as password expiration). The default settings are suitable in most cases.
- **Details.** Select this option to edit details of the user account. The default settings are suitable in most cases.
- **Receive System Mail.** Select this option to forward all emails addressed to root to this user.
- **Automatic Login.** Select this option to enable automatic login for this user. This option logs in the user automatically (without requesting a password) when the system starts.

You should not enable this feature on a production system.
- **User Management.** Select this option to add more users (with the YaST User Management module).



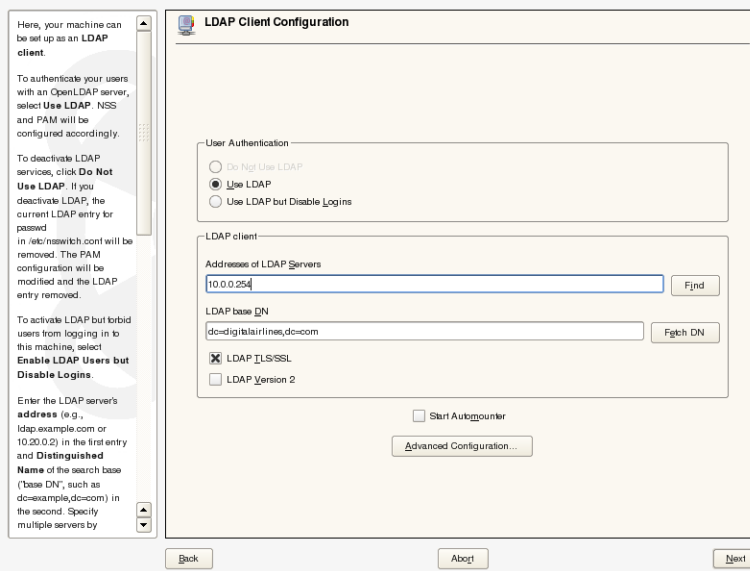
You can add other users later (after installation), but you have to create at least one user during installation so you don't have to work as the user root after the system has been set up.

After you enter all required information, select **Next**.

Configure the System as an LDAP Client

If you select LDAP as authentication method, the following appears:

Figure 1-24



From this dialog, you can configure your system as an LDAP client. The default configuration suggests to use a locally installed LDAP server.

You can change the configuration with the following options:

- **LDAP client.** You can configure the following:
 - **Addresses of LDAP Servers.** Enter the address of the LDAP server.
 - **LDAP base DN.** Enter the search base on the server.

- ❑ **LDAP TSL/SSL.** Select this option to encrypt the communication with the LDAP server.
- ❑ **LDAP Version2.** Select this option if your LDAP server only support LDAP version 2. By default, LDAP version 3 is used.
- **Start Automounter.** If your LDAP server provides information about the automatic mounting of file systems (such as home directories), you can start the automounter and use the automount information from the LDAP server.
- **Advanced Configuration.** Select this option to change advanced LDAP settings.

When finished with the LDAP configuration, select **Next**.

The Release notes are displayed. You should read them to make sure you are informed about the latest changes.

Configure Hardware

Selecting Next opens the Hardware Configuration dialog.

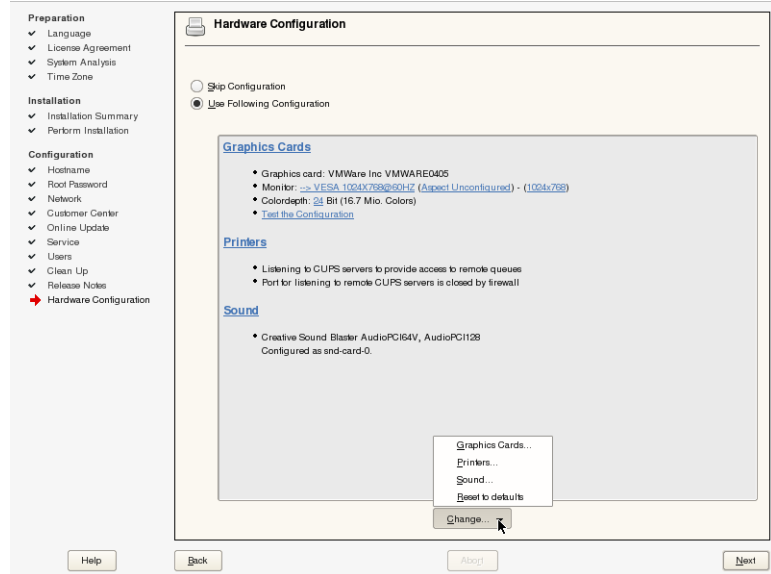
The configuration proposal contains the following items:

- **Graphics Cards.** Displays the graphic card and monitor setup.
- **Printers.** Displays the printer and printer server settings.
- **Sound.** Displays the configuration of the sound card.

To change the automatically generated configuration, select the headline of the item you want to change, or select the corresponding entry in the **Change** drop-down list.

You can also use the **Change** drop-down list to reset all settings to the automatically generated configuration proposal.

Figure 1-25



You can skip the hardware configuration at this time and configure your devices later in the installed system. However, if the settings of the graphics card in the configuration proposal are not correct, you should change them now to avoid problems during the first system start.

This is done by selecting the respective underlined entry and adjusting the values as needed.

Finalize the Installation Process

Confirm your hardware settings by selecting **Next**, and then select **Finish**. Unless you remove the check mark in front of **Clone This System for Autoyast**, an autoyast file is generated and saved as `/root/autoinst.xml`, which you can use to set up an identical system.

The system starts the graphical login screen, where you can log in with your previously created user account. SLES 10 is now installed on your system.

Objective 3 Troubleshoot the Installation Process

SUSE Linux Enterprise Server 10 has been installed and tested on many different machines and hardware platforms. However, sometimes problems can occur.

The following table contains an overview of the most common installation problems, possible causes, and solutions:

Table 1-1

Problem	Possible Cause	Solution
The system does not start from the installation media.	The system is not configured to boot from the CD or DVD drive.	Enter the BIOS setup of the system and choose the CD or DVD drive as the first boot drive. Read the system manual for details about the BIOS setup.
	The CD or DVD drive is defective.	Try to boot a different system with <i>SLES 10 CD 1</i> . If it works, the CD or DVD drive of the actual system might be defective.
	The installation CD or DVD is defective.	If the installation CD does not boot on a different system, the CD or DVD itself could be defective. Contact your reseller to exchange the SLES 10 CD or DVD set.

Table 1-1 *(continued)*

Problem	Possible Cause	Solution
The installation program does not start.	Your system does not support newer hardware features correctly.	Select Installation – ACPI Disabled . If that doesn't fix the problem, select Installation – Save Settings from the Boot menu of the CD or DVD.
	Your system has less than 256 MB of main memory.	Install at least 256 MB of main memory and start the installation again.
The installation process stops.	Your system does not support newer hardware features correctly.	Select Installation – ACPI Disabled . If that doesn't fix the problem, select Installation – Save Settings from the Boot menu of the CD or DVD.
	The installation CD or DVD is defective.	If the installation process also stops on a different system, the CD or DVD could be defective. Contact your reseller to exchange the SLES 10 CD or DVD set.

Table 1-1 *(continued)*

Problem	Possible Cause	Solution
The network connection test or Online Update fails.	There is no DHCP server in the network.	If you configured your network card to use DHCP, assign a static IP address and configure routing and DNS settings manually.
	There is no route to the Internet.	Set the default gateway correctly.
	There is no direct Internet connection, but the system is using no or the wrong proxy settings.	Set the correct proxy configuration in the network configuration dialog. You can also skip the connection test and the Online Update and perform an Online Update in the installed system.
The graphical login does not appear after the installation is completed.	You are using the wrong X11 configuration.	Change to a text console and enter init 3 . Start sax2 from the command line and correct the X11 configuration. Enter init 5 to get a graphical login screen.

Exercise 1-1 Install SUSE Linux Enterprise Server 10

In this exercise, you install SUSE Linux Enterprise Server 10.

You will find this exercise in the workbook.

(End of Exercise)

Summary

Objective	Summary
1. Perform a SLES 10 Installation	<p>During the installation, the hard disks are prepared and the software packages are installed.</p> <p>The following tasks belong to the installation:</p> <ul style="list-style-type: none">■ Boot from the installation media.■ Select the language.■ Select the installation mode.■ Understand and change the installation proposal.■ Perform hard disk partitioning.■ Change the software selection.■ Launch the installation process.
2. Configure the SLES 10 Installation	<p>In the configuration step, you customize and configure the installed system.</p> <p>The following tasks belong to the configuration step:</p> <ul style="list-style-type: none">■ Set the root password.■ Configure the network.■ Configure Network Services.■ Manage Users.■ Configure Hardware.■ Finalize the Installation Process.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
3. Troubleshoot the Installation Process	<p>SLES 10 has been installed and tested on many different machines and hardware platforms. However, sometimes installation problems can occur.</p> <p>Some issues to look for are:</p> <ul style="list-style-type: none">■ The system is not configured to boot from the CD or DVD drive.■ The CD or DVD drive is defective.■ The installation CD or DVD is defective.■ The system does not support newer hardware features (ACPI) correctly.■ There is no DHCP server in the network.■ There is no route to the Internet.■ You are using the wrong proxy settings.■ You are using the wrong X11 configuration.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 2 YaST

YaST (***Yet another Setup Tool***) is not only used during installation, but also for many different tasks during routine system administration.

In RHEL, separate tools (system-config-*) are used for specific purposes. In SUSE Linux Enterprise Server, YaST is the central system management tool. All YaST modules are accessible from the YaST Control Center. However, YaST modules can also be started separately.

This section covers the purpose of YaST and its key modules. Further modules will be covered in the remaining sections of this course.

Objectives

1. Get to Know YaST
2. YaST Software Management
3. Manage User and Group Accounts with YaST
4. Find the YaST Module You Need

Objective 1 Get to Know YaST

To understand YaST, you should be familiar with

- The YaST User Interface
- The Role of SuSEconfig

The YaST User Interface

The YaST user interface can appear in two kinds:

- **ncurses**. Text mode
- **QT**. Fully graphical mode

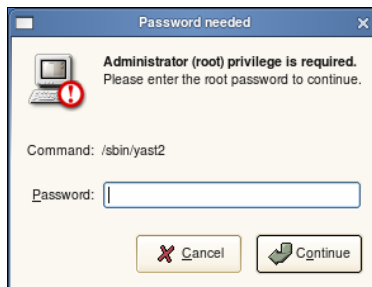
The appearance of the user interface depends on which command you use to start YaST and on whether you use the graphical system or the command line, as indicated in the following:

Table 2-1

Command	Terminal in X Window	Command Line
yast2	Qt	ncurses
yast	ncurses	ncurses

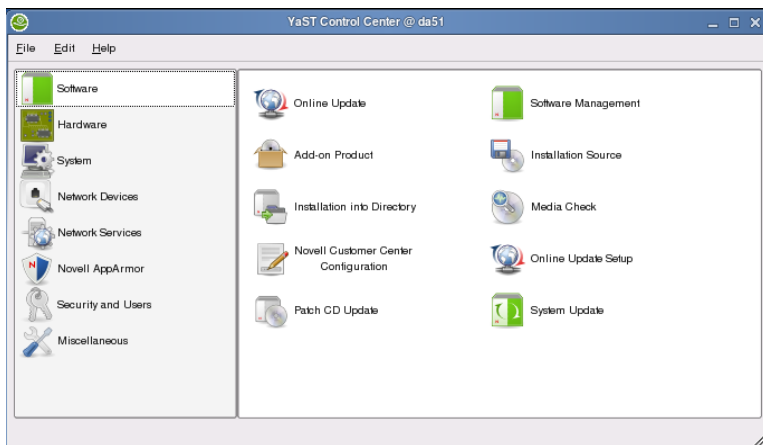
In the graphical interface, you can control YaST with the mouse. To start it, select **YaST** from the main menu (application group: **System**). You are asked to enter the root password.

Figure 2-1



The YaST dialog appears.

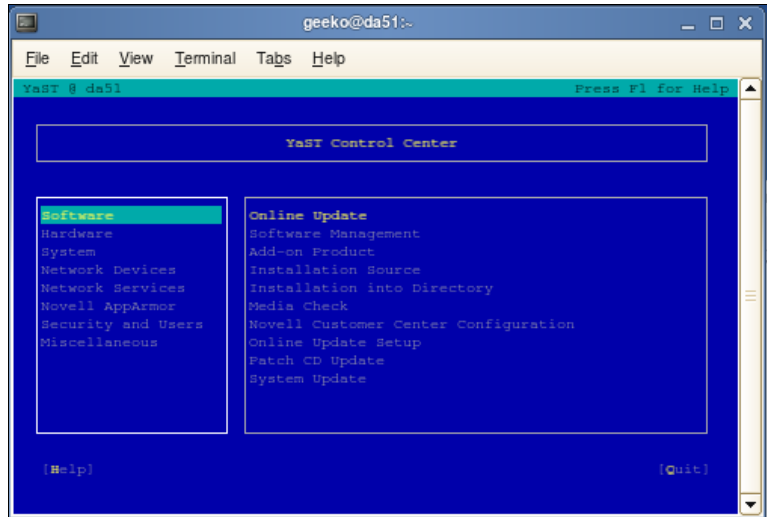
Figure 2-2



You control the ncurses interface with the keyboard. To start the ncurses interface of YaST, you can start a terminal emulation from your GNOME desktop by selecting **Gnome Terminal** from the main menu (application group: **System**).

Enter **su -** to get root permissions. After entering the root password, start YaST by entering **yast**.

Figure 2-3



Press **Tab** to move from one box to another or to the text buttons. To go back to the previous box, press **Shift+Tab**. Use the arrow keys to navigate within the box. Mark highlighted menu items by pressing the **Spacebar**.

To select a menu item, press **Enter**. You can often press **Alt** and the highlighted letter to access an item directly.

Except for the controls and the appearance, the graphical mode and the text mode of YaST are identical.

You can list the available YaST modules with the command **yast -l** or **yast --list**. To start an individual module, specify its name. For example, you can enter the following to start the software installation module:

```
yast sw_single
```

You can enter the software module name with the command **yast** or **yast2**, as in the following:

- **yast sw_single** (text mode)
- **yast2 sw_single** (graphical mode)



To install a software package you can also enter **yast -i *package.rpm***. The package is installed directly without any dialogs.

To display a list of YaST options, enter one of the following:

- **yast --help**
- **yast -h**

The main dialog of YaST is called the *YaST Control Center*.

From here you can select a category on the left (such as Software or System) and a module on the right (such as Online Update) to configure and manage your system.

When you finish making changes with a YaST module, YaST uses back end services such as SuSEconfig.

The Role of SuSEconfig

You can consider YaST as a front end to various other programs, such as a front end to RPM (RPM Package Manager) software management, a front end to user management, or a front end to various configuration files of different services (like a mail or web server).

Usually YaST writes your configuration changes directly into the final configuration file. In rare cases there is an additional intermediate step, where the information you enter is first written to a file in the directory `/etc/sysconfig/` before it is written into the final configuration file.

This is where the program **SuSEconfig** becomes important.

SuSEconfig is a tool used in SUSE Linux Enterprise Server to configure the system according to the variables that are set in the various files in `/etc/sysconfig/` and its subdirectories.

These files contain variables such as `SYSLOGD_PARAMS=""` in `/etc/sysconfig/syslog` and `SMTPD_LISTEN_REMOTE="no"` in `/etc/sysconfig/mail`.

Some of these variables are used directly (such as in some start scripts). For example, if `SYSLOGD_PARAMS` is set to `"-r"`, the daemon that logs system messages is directed to listen on port 514 for system messages from other hosts.

Other variables are used to modify other files. For example, if `SMTPD_LISTEN_REMOTE` is set to `"yes"`, the variable `INET_INTERFACES` in `/etc/postfix/main.cf` is set to `"all"` by the script `/sbin/SuSEconfig` and the scripts in `/sbin/conf.d/`.

SuSEconfig acts as a back end for YaST and activates the configuration changes you make when using a YaST module.

If you modify files in `/etc/sysconfig/` using an editor, all you might need to do is restart a service for the change to take place. However, you might also need to run SuSEconfig.

For this reason, we recommend running SuSEconfig after manually editing files in `/etc/sysconfig/`.

SuSEconfig uses the subsystem specific scripts in `/sbin/conf.d/` to configure the various subsystems. For example the variables in `/etc/sysconfig/postfix` are evaluated by the script `/sbin/conf.d/SuSEconfig.postfix`.

Exercise 2-1 *Get to Know YaST*

In this exercise, you learn how to use the different user interfaces of YaST and how to start some YaST modules.

You will find this exercise in the workbook.

(End of Exercise)

Objective 2 YaST Software Management

Software management has different aspects:

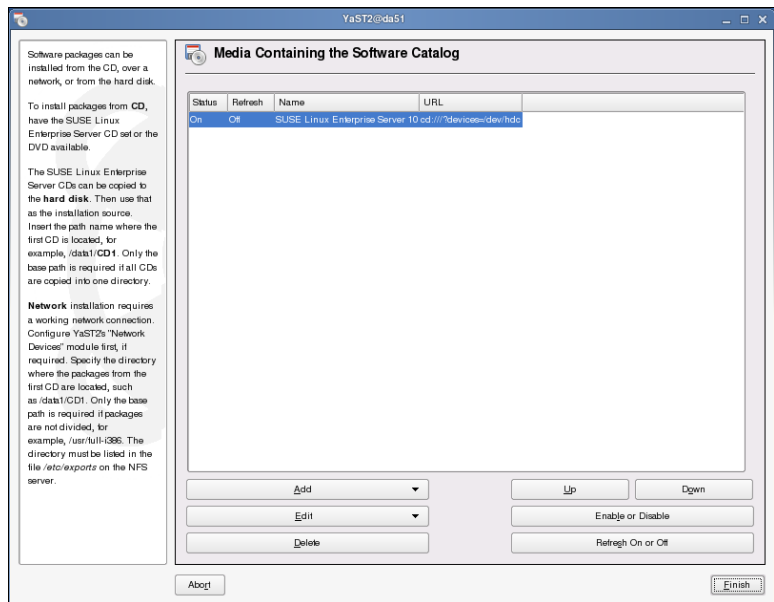
- Manage Installation Sources
- Install Software Packages
- Apply Security Updates

Manage Installation Sources

The software installation dialog lists only the packages that are on the current installation media.

If you want to add more installation sources, select **Software > Installation Source** from the YaST Control Center. The following appears:

Figure 2-4



1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

To add a new source, select the **Add** drop-down list and select the type of installation source. Depending on the type of source, you might have to provide additional information (such as the IP address of an installation server).

To edit the configuration of an existing installation source, select the source in the list and select **Edit**.

If you want to disable an installation source temporarily, select the source in the list and select **Enable or Disable**.

To remove an installation source permanently, select the source from the list and select **Delete**.

YaST uses the first installation source in the list that has the software package you want to install. To change the order of a source in the list, select the source and **Up** or **Down**.

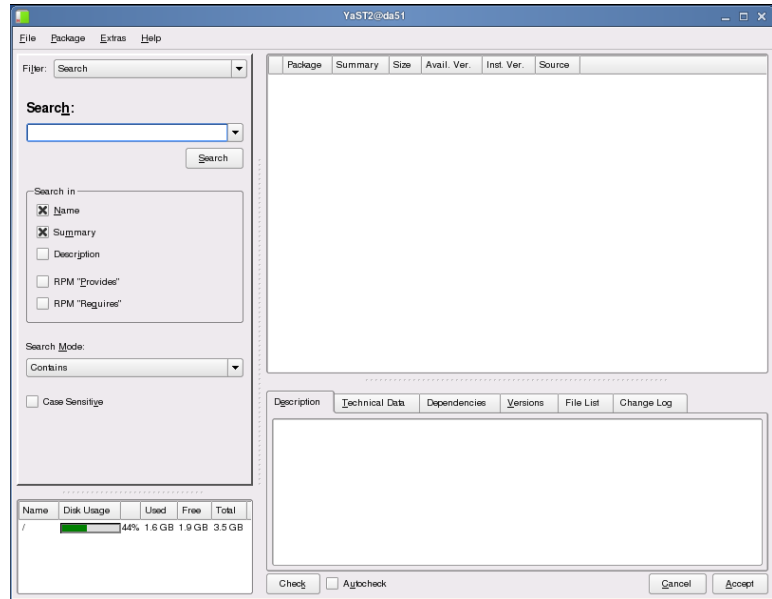
Install Software Packages

After performing a standard SUSE Linux Enterprise Server 10 installation, you will often need to install additional software. To do this, start the YaST module **Software > Software Management**.

The installed packages and the packages that are available on the installation media are analyzed and dependencies between packages are checked.

After this checking, the following dialog for searching packages appears:

Figure 2-5



To help you find the software you want to install, you can choose from different filters listed in the drop-down list in the top left corner of the window labeled **Filter**. The following filters are available:





- **Patterns.** Displays all software that is available on the known installation media. It is grouped in predefined sets of packages that logically belong together.
- **Package Groups.** Displays all software that is available on the known installation media. It is grouped by category.
- **Languages.** Displays all language related files (e.g., spell checker, help)
- **Installation Sources.** Lists all registered installation sources and displays the available packages of this source.

- **Search.** Lets you enter a search term and where you want YaST to search for the software package.
- **Installation Summary.** Displays all the packages with a marked status.

To find a package, select the Search filter, enter the package name or parts of the package name or some keywords in the Search field; then select **Search**.

The matched packages are listed in the right area. The installation state is shown by a small symbol in front of the package name. The most commonly displayed symbols include the following:

Figure 2-6

<input type="checkbox"/>	Do not install	This package is not installed and it will not be installed.
<input checked="" type="checkbox"/>	Install	This package will be installed. It is not installed yet.
<input checked="" type="checkbox"/>	Keep	This package is already installed. Leave it untouched.
	Update	This package is already installed. Update it or reinstall it (if the versions are the same).
	Delete	This package is already installed. Delete it.
	Taboo	This package is not installed and should not be installed under any circumstances, especially not because of unresolved dependencies that other packages might have or get. Packages set to "taboo" are treated as if they did not exist on any installation media.
	Protected	This package is installed and should not be modified, especially not because of unresolved dependencies that other packages might have or get. Use this status for third-party packages that should not be overwritten by newer versions that may come with the distribution.
<input checked="" type="checkbox"/>	Autoinstall	This package will be installed automatically because some other package needs it. Hint: You may have to use "taboo" to get rid of such a package.

To view a list of all possible symbols, select **Help > Symbols**.

Select the symbol of the package you want to install several times until the Install symbol appears; then select **Accept**.

You might see a dialog indicating that the dependencies between the packages cannot be solved and that some other packages need to be installed, too. In most cases you can simply confirm this dialog.

If the wrong CD or DVD is in your drive, a warning appears.

Apply Security Updates

SUSE Linux Enterprise Server 10 is sold with system maintenance. This system maintenance includes updates and security patches.

Software updates can be managed with the YaST Online Update (YOU) module. This YaST module downloads and installs software updates and security patches.

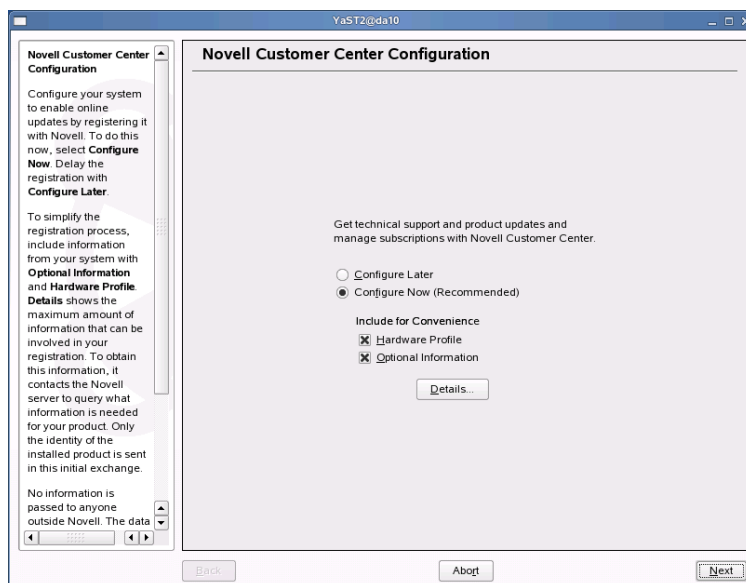
To apply security updates, you need to do the following:

- Configure the Novell Customer Center
- Use the YaST Online Update

Configure the Novell Customer Center

To access the configuration of the Novell Customer Center, start the YaST Control Center and select **Software > Novell Customer Center Configuration**. You can also start the dialog directly from a terminal window as root by entering `yast2 inst_suse_register`. The dialog is the same as that offered during installation for this purpose:

Figure 2-7



Selecting Details shows what information is being collected and sent.

With a browser, the Novell Customer Center can be accessed at <http://www.novell.com/center/>. After you have created a Novell account, you need to register your product with the registration code delivered with the SUSE Linux Enterprise Server 10 product.



The SUSE Linux Enterprise Server 10 DVD you received as part of your student kit does not include maintenance. Visit the Novell Customer Center for information on how to take part in the SUSE Linux Enterprise Server 10 Maintenance Program.

Only registered products with a valid maintenance contract can be updated with the YOU module.

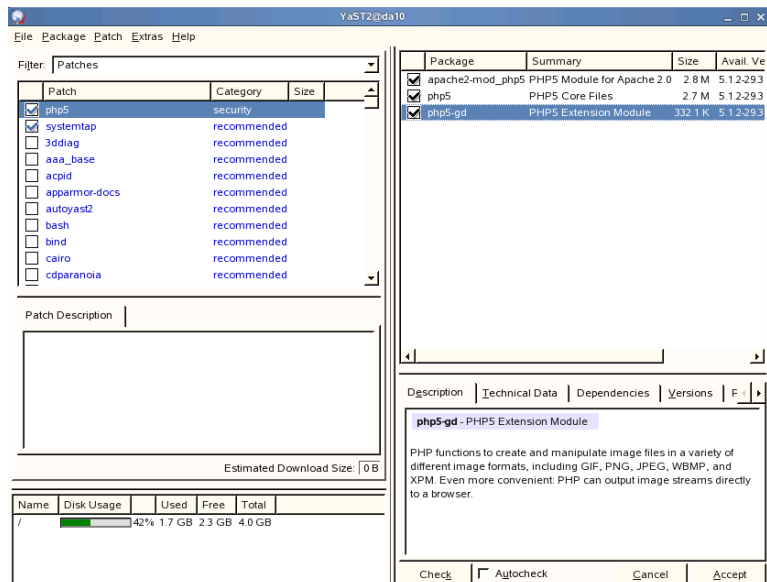
Use the YaST Online Update

The following is a quick guide to applying software updates with YOU.

Start the YOU module from the YaST Control Center by selecting **Software > Online Update**.

The following appears:

Figure 2-8



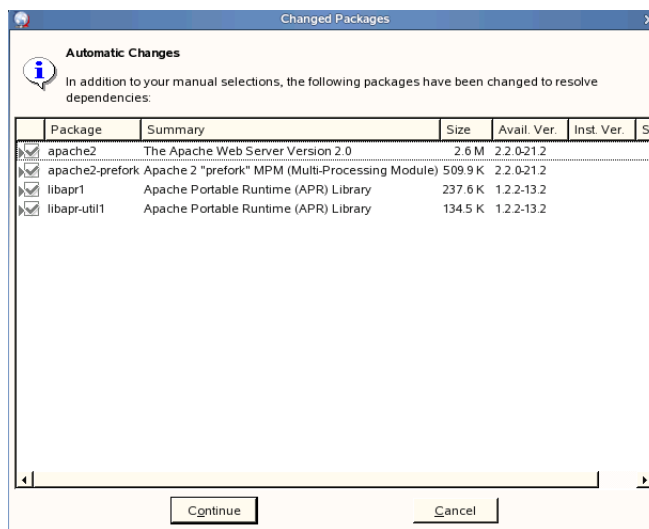
1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

On the top left side of the dialog all available patches are displayed. Select an entry to see details for the update on the right side of the dialog. To have an update installed in the next step, select the check box next to the corresponding entry.

Select **Accept** to start the update process.

Depending on your selection an **Automatic Changes** dialog appears:

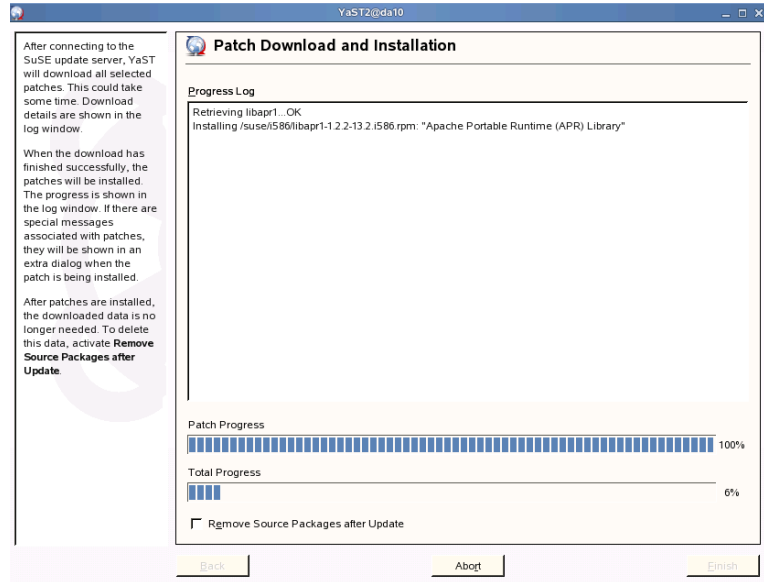
Figure 2-9



Accept the changes by selecting **Continue**; the patches are transferred from the update server and installed.

During the installation process YOU displays the following dialog:

Figure 2-10



Sometimes additional information is displayed for some updates. These dialogs need to be confirmed to install the corresponding software package.

Once all updates have been downloaded and installed, select **Finish** to close the dialog.

Exercise 2-2 *Install New Software*

In this exercise, you install another software package that is available on the SUSE Linux Enterprise Server 10 installation media.

You will find this exercise in the workbook.

(End of Exercise)

Objective 3 **Manage User and Group Accounts with YaST**

With YaST, you can manage users and groups. To do this, you need to understand the following:

- Basics about Users and Groups
- User and Group Administration with YaST

Basics about Users and Groups

One of the main characteristics of a Linux operating system is its ability to handle several users at the same time (multiuser) and to allow these users to perform several tasks on the same computer simultaneously (multitasking).

For this reason the system must be able to uniquely identify all users. To achieve this, every user must log in with the following:

- A username
- A password

Because the operating system can handle numbers much better than strings, users are handled internally as numbers. The number which a user receives is a *UID* (User ID).

Every Linux system has a privileged user, the user *root*. This user always has the UID 0. This is the administrator of the system.

Users can be grouped together based on shared characteristics or activities. For example:

- On SUSE Linux Enterprise Server 10, users by default have the group *users* as their primary group. (This is different from RHEL, where each user has a group of his own as his primary group.)

- All users who intend to create web pages can be placed in the group *webedit*.

Of course, file permissions for the directory in which the web pages are located must be set so that the group *webedit* is able to write (save files) in that directory.

As with users, each group is also allocated a number internally called the *GID* (Group ID), and can be one of the following types:

- Normal groups
- Groups used by the system
- The root group (GID = 0)

User and Group Administration with YaST

You can access YaST user and group account administration in the following ways:

- User Administration

From the YaST Control Center, select **Security and Users > User Management**, or from a terminal window, enter **yast2 users**.

- Group Administration

From the YaST Control Center, select **Security and Users > Group Management**, or from a terminal window, enter **yast2 groups**.

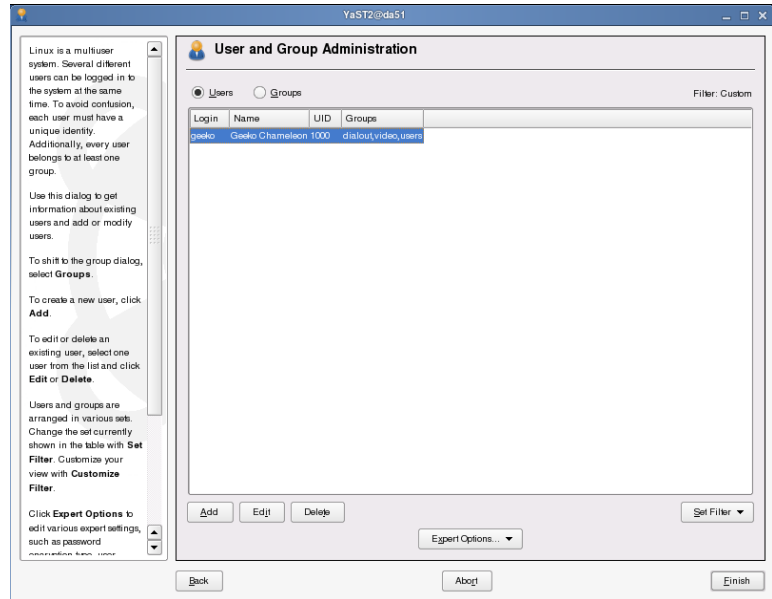
If you have selected LDAP for authentication during the installation of the SUSE Linux Enterprise Server, you are prompted for the LDAP server administrator password.

You can switch back and forth between administering users and administering groups by selecting the **Users** and **Groups** radio buttons at the top of the module window.

User Administration

The user account management window lists the existing user accounts (as in the following):

Figure 2-11



A list of users (accounts on your server) appears with information such as login name, full name, UID, and associated groups included for each user.

Select **Set Filter**; then select one of the following to change the users listed:

- **Local Users.** User accounts you have created on your local server for logging into the server.
- **System Users.** User accounts created by the system for use with services and applications.

- **Custom.** A customized view of users based on the settings configured with **Customize Filter**.
- **Customize Filter.** This option lets you combine listed user sets (such as **Local Users** and **System Users**) to display a customized view (with **Custom**) of the users list.

Additional sets of users (such as **LDAP users**) are added to the **Set Filter** drop-down list as you configure and start services on your server.

You can create a new user account or edit an existing account by selecting **Add** or **Edit**.

The following appears:

Figure 2-12

YaST2@da51

Existing Local User

User Data Details Password Settings

Enter the User's Full Name, Username, and Password to assign to this user account.

When entering a password, distinguish between uppercase and lowercase. Passwords should not contain any special characters, such as accented characters.

With the current password encryption (Blowfish), the password length should be between 5 and 72 characters.

Valid password characters are letters, digits, blanks, and `!@#$%^&*~.-_+=`.
`$%& / > { [()] =`

To ensure that the password was entered correctly, repeat it exactly in a second field. Do not forget your password.

For the Username, use only letters (no accented characters), digits, and `_-.`. Do not use uppercase letters in this entry unless you know what you are doing. Usernames have stricter restrictions than passwords. You can

User's Full Name
Geeko Chameleon

Username
geeko

Password

Confirm Password

☐ Disable User Login

Cancel Accept

Enter or edit information in the following fields:

- **User's Full Name.** Enter a real user name (such as **Geeko Chameleon**)
- **Username.** Enter a user name that is used to log in to the system (such as **geeko**).
- **Password and Confirm Password.** Enter and re-enter a password for the user account.

When entering a password, distinguish between uppercase and lowercase letters.

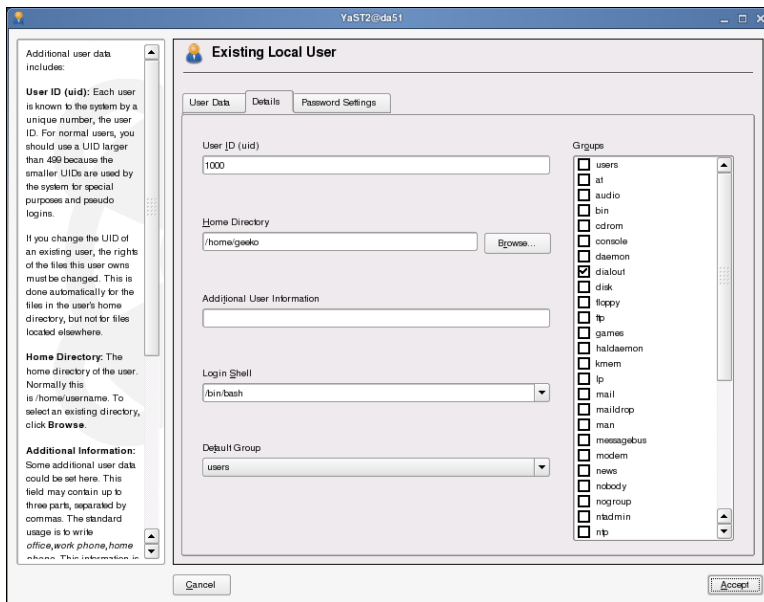
Valid password characters include letters, digits, blanks, and #*,.,:;_~+!\$%&/'?{[()]}=.

The password should not contain any special characters (such as accented characters), as you might find it difficult to type these characters on a different keyboard layout when logging in from another country.

With the current password encryption (Blowfish), the password length should be between 5 and 72 characters.

To set the properties of the user (such as the UID, the home directory, the login shell, group affiliation, and additional user account comments), select the **Details** tab. The following appears:

Figure 2-13



Enter or edit information in the following fields:

- **User ID (uid).** For normal users, you should use a UID greater than 999 because the lower UIDs are used by the system for special purposes and pseudo logins.

If you change the UID of an existing user, the permissions of the files of this user owns must be changed. This is done automatically for the files in the user's home directory, but not for files located elsewhere.



If this does not happen automatically, you can change the permissions of the user files in the home directory (as root) by entering **chown -R username /home/username**.

- **Home Directory.** The home directory of the user. Normally this is **/home/username**.

You can select an existing directory by selecting **Browse**.

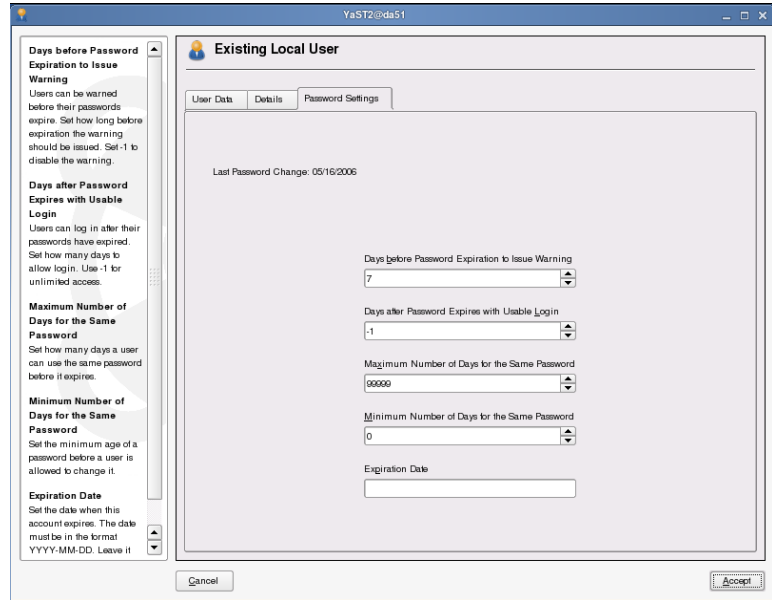
- **Additional User Information.** This field can contain up to 3 parts separated by commas. It is often used to enter *office,work phone,home phone*.

This information is displayed when you use the **finger** command on this user.

- **Login Shell.** From the drop-down list select the default login shell for this user from the shells installed on your system.
- **Default Group.** This is the group to which the user belongs. Select a group from the list of all groups configured on your system.
- **Groups.** Select all additional memberships you want to assign to the user from the list.

To set various password parameters (such as duration of a password), select the **Password Settings** tab. The following appears:

Figure 2-14



Enter or edit information in the following fields:

- **Days before Password Expiration to Issue Warning.** Enter the number of days before password expiration that a warning is issued to users.

Enter **-1** to disable the warning.

- **Days after Password Expires with Usable Login.** Enter the number of days after the password expires that users can continue to log in.

Enter **-1** for unlimited access.

- **Maximum number of days for the same password.** Enter the number of days a user can use the same password before it expires.
- **Minimum number of days for the same password.** Enter the minimum age of a password before a user can change it.
- **Expiration date.** Enter the date when the account expires. The date must be in the format YYYY-MM-DD.

Leave the field empty if the account never expires.

Save the settings for the new or edited user by selecting **Accept**.

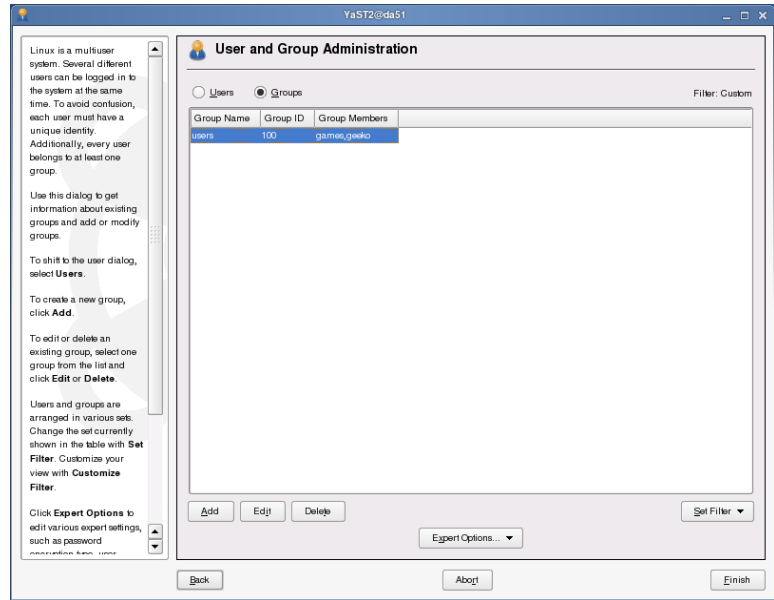
A new user appears in the list.

Configure your server with the new settings by selecting **Finish**.

Group Administration

You can administer groups from the following window:

Figure 2-15



A list of groups appears with information such as group name, Group ID (*GID*), and group members.

Select **Set Filter**; then select one of the following to change the groups listed:

- **Local Groups.** Groups created on your local server to provide permissions for members assigned to the group.
- **System Groups.** Groups created by the system for use with services and applications.
- **Custom.** A customized view of groups based on the settings configured with **Customize Filter**.

- **Customize Filter.** This option lets you combine listed group sets (such as **Local Groups** and **System Groups**) to display a customized view (with **Custom**) of the groups list

Additional sets of groups are added to the **Set Filter** drop-down list (such as LDAP) as you configure and start services on your server.

You can create a new group or edit an existing group by selecting **Add** or **Edit**. The following appears when you select **Edit**:

Figure 2-16

The screenshot shows the 'Existing Local Group' dialog box in YaST2. On the left, there is a sidebar with instructions for Group Name, Group ID (gid), Password, and Confirm Password. The main area contains fields for Group Name (set to 'users'), Group ID (gid) (set to '100'), Password, and Confirm Password. To the right, there is a list of Group Members with checkboxes for 'at', 'bin', 'daemon', 'lp', 'gdm', 'haldaemon', 'lp', 'mail', 'man', 'messagebus', 'news', and 'nobody'. Below this, there is a list of services with checkboxes for 'games' and 'gnome'. At the bottom, there are 'Cancel' and 'Accept' buttons.

Enter or edit information in the following fields:

- **Group Name.** The name of the group. Avoid long names. Normal name lengths are between two and eight characters.

- **Group ID (gid).** The GID number assigned to the group. The number must be a value between 0 and 60000. GIDs to 99 represent system groups. GIDs beyond 99 can be used for normal users. YaST warns you if you try to use a GID that is already in use.

- **Password** (optional). Require the members of the group to identify themselves while switching to this group (see **man newgrp**). To do this, assign a password.

For security reasons, the password is represented by asterisks (“*”).

- **Confirm Password.** Enter the password a second time to avoid typing errors.
- **Group Members.** Select which users should be members of this group.

A second list appears (when you select Edit) that shows users for which this group is the default group. This list cannot be edited from YaST.

When you finish entering or editing the group information, select **Next**. You are returned to the Group Administration dialog. Save the configuration settings by selecting **Finish**.

The information you enter when creating or editing users and groups with YaST is saved to the following user administration files:

- /etc/passwd
- /etc/shadow
- /etc/group

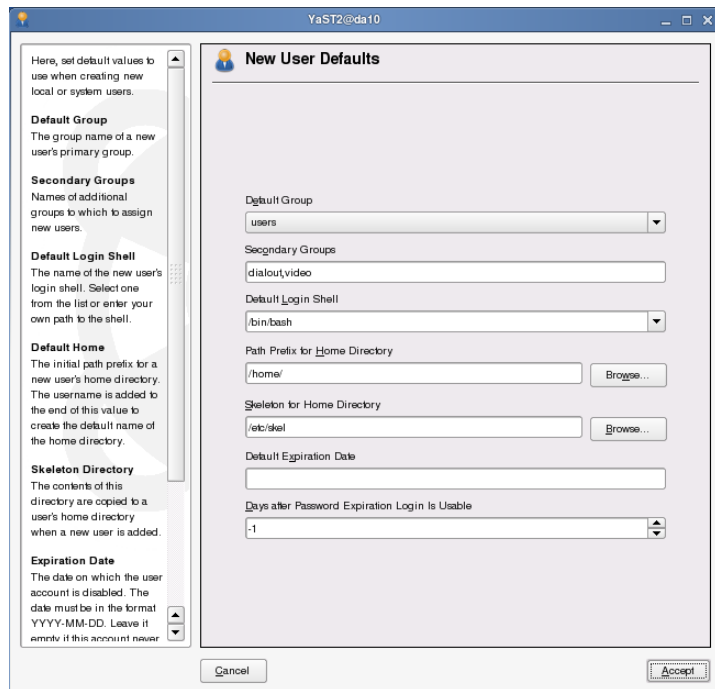
Set Defaults for New User Accounts

You can use YaST to select default settings to be applied to new user accounts.

From the Gnome desktop, press **Alt+F2**, enter **yast2** and enter the root password when prompted. Select **Security and Users > User Management**. You can also start the User Management module directly from a terminal window as root by entering **yast2 users**.

Select **Expert Options > Defaults for New Users**. The following appears:

Figure 2-17



You can enter or edit information in the following fields:

- **Default Group.** From the drop-down list select the primary (default) group.
- **Secondary Groups.** Enter a list of secondary groups (separated by commas) to assign to the user.
- **Default Login Shell.** From the drop-down list select the default login shell (command interpreter) from the shells installed on your system.
- **Default Home.** Enter or browse to the initial path prefix for a new user's home directory. The user's name will be appended to the end of this value to create the default name of the user's home directory.
- **Skeleton Directory.** Enter or browse to the skeleton directory. The contents of this directory will be copied to the user's home directory when you add a new user.
- **Default Expiration Date.** Enter the date on which the user account is disabled. The date must be in the format YYYY-MM-DD.

Leave the field empty if this account never expires.
- **Days after Password Expiration Login Is Usable.** This setting enables users to log in after passwords expire. Set how many days login is allowed after a password expired.

Enter **-1** for unlimited access.

Save the configuration settings by selecting **Next > Finish**. The values are written to the file **/etc/default/useradd**:

```
da10:~ # cat /etc/default/useradd
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
GROUPS=video,dialout
CREATE_MAIL_SPOOL=no
```

You can also use the command line program **useradd** to view or change the defaults. The option **--show-defaults** displays the same as the cat above. The option **--save-defaults** followed by an option with a value changes them:

```
da10:~ # useradd --save-defaults -d /export/home
da10:~ # useradd --show-defaults
GROUP=100
HOME=/export/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
GROUPS=video,dialout
CREATE_MAIL_SPOOL=no
```

The manual page for **useradd** lists the possible options.



useradd on SUSE Linux Enterprise Server 10 does not create a home directory for a new user by default; use the option **-m** to create a home directory when creating a new user.

Configure Security Settings

YaST provides a Local Security module that lets you configure the following local security settings for your SUSE Linux Enterprise Server:

- Password settings
- Boot configuration
- Login settings
- User creation settings
- File permissions

You can select from (or modify) three preset levels of security, or create your own customized security settings to meet the requirements of your enterprise security policies and procedures.

You can access the Security Settings module from the YaST Control Center by selecting **Security and Users > Local Security**, or by entering as root **yast2 security** in a terminal window.

The following appears:

Figure 2-18



From this dialog, you can select one of the following preset configurations:

- **Home Workstation.** Select for a home computer not connected to any type of a network. This option represents the lowest level of local security.
- **Networked Workstation.** Select for a computer connected to any type of a network or the Internet. This option provides an intermediate level of local security.
- **Network Server.** Select for a computer that provides any type of service (network or otherwise). This option enables a high level of local security.
- You can also select **Details** or **Custom Settings** to modify an existing security level or create your own configuration.

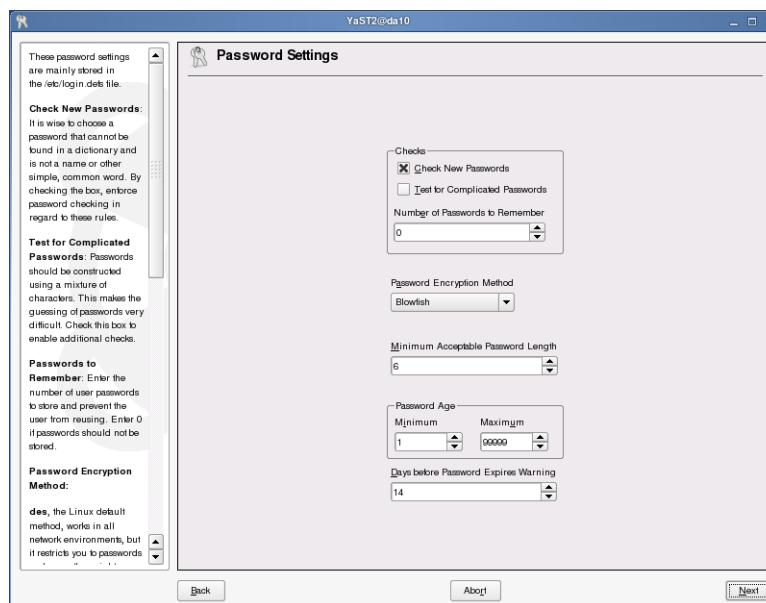
By selecting one of the three predefined security levels and selecting **Next**, the chosen security level is applied. By selecting **Details**, you can change the settings for the security level you have selected.

If you choose the **Customs Settings** and then select **Next**, you can directly change the details of the security configuration.

The dialogs for the detail settings look the same for every security level, but the preselected options are different. In the following dialogs, you see the settings for Level 3 (Network Server).

In the first dialog you can change the default password requirements that are accepted by the systems:

Figure 2-19



From this dialog, you can select or enter the following password settings (mainly stored in `/etc/login.defs`, some values also in `/etc/default/passwd` and `/etc/security/pam_pwcheck.conf`):

- **Check New Passwords.** It is important to choose a password that cannot be found in a dictionary and is not a name or other simple, common word. By selecting this option, you enforce password checking in regard to these rules.
- **Test for Complicated Passwords.** Passwords should be constructed using a mixture of uppercase and lower case characters as well as numbers. Special characters like ;(= etc. may be used too, but could be hard to enter on a different keyboard layout. This makes it very difficult to guess the password. Select this option to enable additional checks.
- **Password Encryption Method.** From the drop-down list, select one of the following encryption methods:
 - **DES.** This is the lowest common denominator. It works in all network environments, but it restricts you to passwords no longer than eight characters. If you need compatibility with other systems, select this method.
 - **MD5.** This encryption method allows longer passwords and is supported by all current Linux distributions, but not by other systems or older software.
 - **Blowfish.** This encryption method uses the blowfish algorithm to encrypt passwords. It is not yet supported by many systems. A lot of CPU power is needed to calculate the hash, which makes it difficult to crack passwords with the help of a dictionary. It is used as default encryption method on SLES 10
- **Minimum Acceptable Password Length.** Enter the minimum number of characters for an acceptable password. If a user enters fewer characters, the password is rejected.

Entering **0** disables this check.
- **Password Age.** Minimum refers to the number of days that have to elapse before a password can be changed again. Maximum is the number of days after which a password expires and must be changed.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

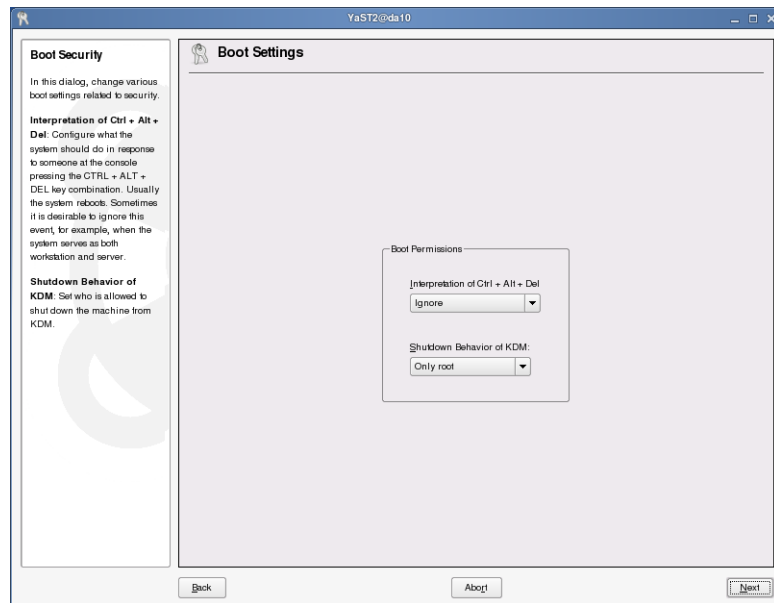
- **Days Before Password Expires Warning.** A warning is issued to the user this number of days before password expiration.



Although root receives a warning when setting a password, she can still enter a bad password despite the above settings.

When you finish configuring password settings, continue by selecting **Next**. The following appears:

Figure 2-20



From this dialog, you can select the following boot settings (which update the file `/etc/inittab`):

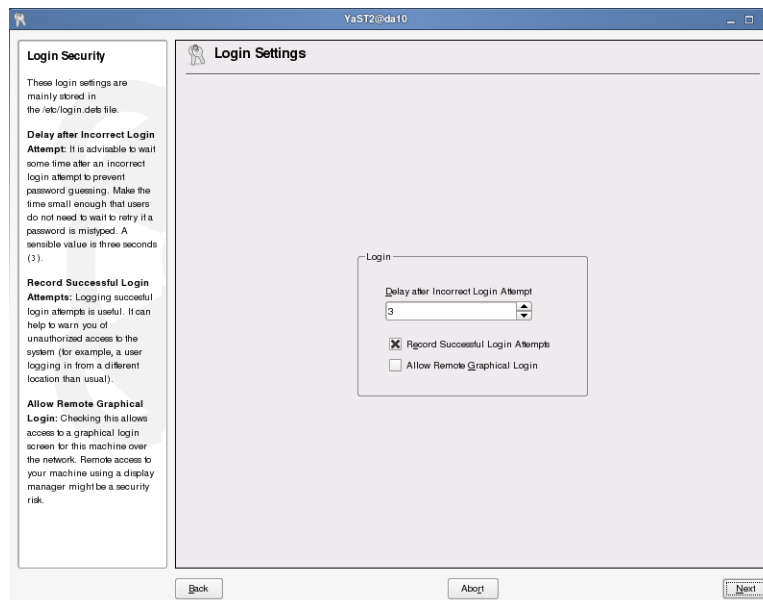
- **Interpretation of Ctrl + Alt + Del.** When someone at the console presses the **Ctrl+Alt+Del** keystroke combination, the system usually reboots.

- ❑ **Ignore.** Sometimes you want to have the system ignore this keystroke combination, especially when the system serves as both workstation and server. Nothing happens when the **Ctrl+Alt+Del** keystroke combination is pressed.
- ❑ **Reboot.** The system reboots when the **Ctrl+Alt+Del** keystroke combination is pressed.
- ❑ **Halt.** The system is shut down when the **Ctrl+Alt+Del** keystroke combination is pressed.
- **Shutdown Behavior of KDM.** You use this option to set who is allowed to shut down the computer from KDM.
 - ❑ **Only Root.** To halt the system, the root password has to be entered.
 - ❑ **All Users.** Everyone, even remotely connected users, can halt the system using KDM.
 - ❑ **Nobody.** Nobody can halt the system with KDM.
 - ❑ **Local Users.** Only locally connected users can halt the system with KDM.
 - ❑ **Automatic.** The system is halted automatically after log out.

For a server system you should use **Only Root** or **Nobody** to prevent normal or even remote users from halting the system

When you finish configuring boot settings, continue by selecting **Next**. The following appears:

Figure 2-21



From this dialog, you can enter and select the following login settings (mainly stored in `/etc/login.defs`):

- **Delay After Incorrect Login Attempt.** Following a failed login attempt, there is typically a waiting period of a few seconds before another login is possible. This makes it more difficult for password crackers to log in.

This option lets you adjust the time delay before another login attempt. Default is 3 seconds, which is a reasonable value.

- **Record Successful Login Attempts.** Recording successful login attempts can be useful, especially in warning you of unauthorized access to the system (such as a user logging in from a different location than normal).

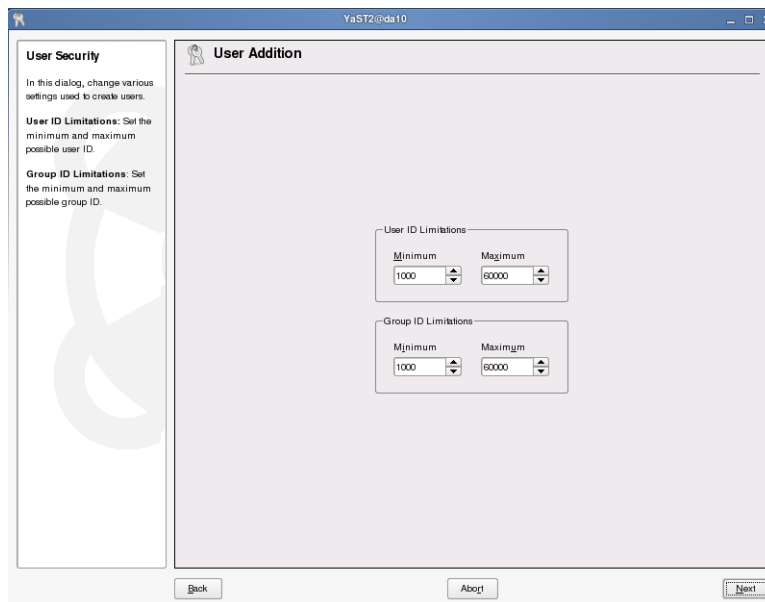
Select this option to record successful login attempts in the file **/var/log/wtmp**. You can use the command **last** to view who logged in at what time.

- **Allow Remote Graphical Login.** You can select this option to allow other users access to your graphical login screen via the network.

Because this type of access represents a potential security risk, it is inactive by default.

When you finish configuring login settings, continue by selecting **Next**. The following appears:

Figure 2-22

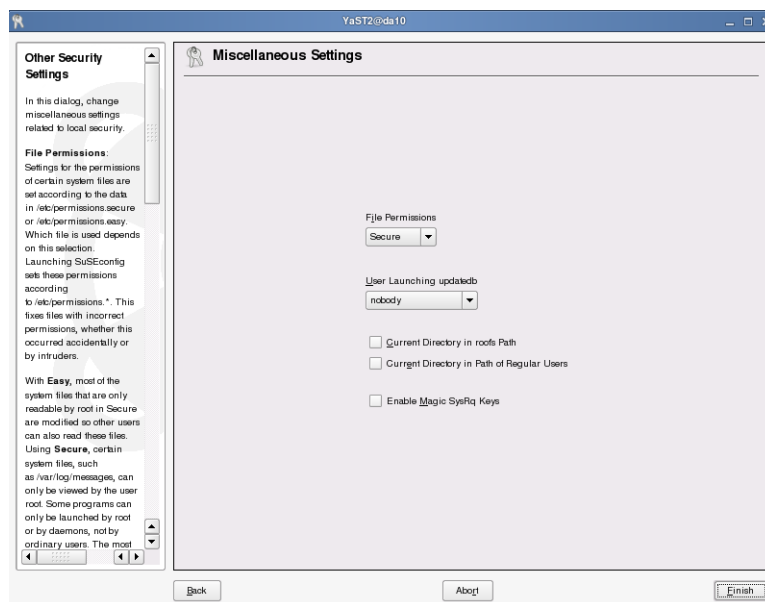


From this dialog, you can enter the following ID settings (stored in **/etc/login.defs**):

- **User ID Limitations.** Enter a minimum and maximum value to configure a range of possible user ID numbers. New users get a UID from within this range.
- **Group ID Limitations.** Enter a minimum and maximum value to configure a range of possible group ID numbers.

When you finish configuring user and group ID limitations, continue by selecting **Next**. The last page of the security configuration appears:

Figure 2-23



From this dialog, you can select the following miscellaneous global settings:

- **File Permissions.** Settings for the permissions of certain system files are configured in **/etc/permissions.easy**, **/etc/permissions.secure**, or **/etc/permissions.paranoid**. You can also add your own rules to the file **/etc/permissions.local**. Each file contains a description of the file syntax and purpose of the preset.

Settings in files in the directory **/etc/permissions.d/** are included as well. This directory is used by packages that bring their own permissions files.

From the drop-down list, select one of the following:

- **Easy.** Select this option to allow read access to most of the system files by users other than root.
- **Secure.** Select this option to make sure that certain configuration files (such as **/etc/ssh/sshd_config**) can only be viewed by the user root. Some programs can only be launched by root or by daemons, not by an ordinary user.
- **Paranoid.** Select this option for an extremely secure system. All SUID/SGID-Bits on programs have been cleared. Remember that some programs might not work correctly, because users no longer have the permissions to access certain files.

Running SuSEconfig sets these permissions according to the settings in the respective **/etc/permissions*** files. This fixes files with incorrect permissions, whether this occurred accidentally or by intruders.

- **User Launching updatedb.** If the program **updatedb** is installed, it automatically runs on a daily basis or after booting. It generates a database (**locatedb**) in which the location of each file on your computer is stored.

You can search this database with the utility **locate** (enter **man locate** for details).

From the drop-down list, select one of the following:

- ☐ **nobody.** Any user can find only the paths in the database that can be seen by any other (unprivileged) user.
- ☐ **root.** All files in the system are added into the database.
- **Current Directory in root's Path and Current Directory in the Path of Regular Users.**

If you deselect these options (the default), users must always launch programs in the current directory by adding “./” (such as **./configure**).

If you select these options, the dot (“.”) is appended to the end of the search path for root and users, allowing them to enter a command in the current directory without appending “./”.

Selecting these options can be very dangerous because users can accidentally launch unknown programs in the current directory instead of the usual system-wide files.

This configuration is written to **/etc/sysconfig/suseconfig**.

- **Enable Magic SysRq Keys.** Selecting this option gives you some control over the system even if it crashes (such as during kernel debugging). For details, see **/usr/src/linux/Documentation/sysrq.txt**.

This configuration is written to **/etc/sysconfig/sysctl**.

When you finish configuring the miscellaneous settings, save the settings and run SuSEconfig by selecting **Finish**.

Exercise 2-3 Manage User Accounts with YaST

In this exercise, you create and remove an user account with the YaST user management module.

You will find this exercise in the workbook.

(End of Exercise)

Exercise 2-4 Configure the Password Security Settings

In this exercise, you practice changing different security settings.

You will find this exercise in the workbook.

(End of Exercise)

Objective 4 Find the YaST Module You Need

Being familiar with the RHEL `system-config-*` tools, you possibly look for the corresponding tool in SUSE Linux Enterprise Server 10.

We compiled a table that lists the `system-config-*` command on the left and the comparable YaST module on the right. You can access the module via the YaST Control Center or you can enter the command given in the right column in a console window.

The YaST modules do not match the respective `system-config-*` tool exactly, but you will most likely find that they cover the functionality pretty well, and often offer more configuration options.

There are many more YaST modules than those listed here. To view the complete list, enter `yast -l` in a console window.

Table 2-2

RHEL	SLES 10
system-config-authentication	yast2 nis yast2 ldap yast2 kerberos-client yast2 samba-client
system-config-date	yast2 timezone
system-config-time	yast2 ntp-server
(both are links to /usr/bin/consolehelper)	
system-config-display	yast2 x11 sax2
system-config-httpd	yast2 http-server
system-config-keyboard	yast2 keyboard
system-config-language	yast2 language

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Table 2-2

RHEL	SLES 10
system-config-lvm	yast2 lvm_config
system-config-printer	yast2 printer
system-config-printer-gui	
system-config-printer-tui	yast printer
system-config-nfs	yast2 nfs_server
system-config-packages	yast2 sw_single
system-install-packages package.rpm	yast2 -i package
system-config-mouse	yast2 mouse
system-config-network-druid	Start YaST Control Center and on the left select Network Devices
system-config-network	yast2 lan
system-config-network-gui	
system-config-network-tui	yast lan
system-config-network-cmd	hwinfo --network --netcard
system-config-rootpassword	yast2 users, select systemusers in the Filter drop-down menu, then select root and change the password
system-config-samba	yast2 samba-server
system-config-soundcard	yast2 sound
system-config-securitylevel	yast2 firewall
system-config-securitylevel-tui	
system-config-services	yast2 runlevel

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Table 2-2

RHEL	SLES 10
system-config-users	yast2 users
	yast2 groups

Summary

Objective	Summary
1. Get to Know YaST	<p>The appearance of the user interface of YaST depends on the command used for starting:</p> <ul style="list-style-type: none">■ In the graphical interface, YaST can be controlled intuitively with the mouse.■ The ncurses interface is controlled exclusively with the keyboard. <p>Individual modules can also be started directly. Available modules can be listed with the command yast -l or yast --list.</p> <p>SuSEconfig is a tool used in SUSE Linux Enterprise Server to configure the system according to the variables that are set in the various files in /etc/sysconfig/ and its subdirectories.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
2. YaST Software Management	<p>To add a new installation source, start the YaST module Software > Installation Source.</p> <p>To install new software packages use the YaST module Software > Software Management.</p> <p>The installation status of a package is indicated by a symbol. An overview about all possible symbols can be reached via the Help > Symbols menu.</p> <p>The YaST Online Update and the Novell Customer Center offer an easy way to keep software up-to-date.</p>
3. Manage User and Group Accounts with YaST	<p>Each user account is assigned a unique internal number: the UID (UserID).</p> <p>Every Linux system has a privileged user, the user root. This user always has the UID 0.</p> <p>As with users, the groups are also allocated a number internally: the GID (GroupID).</p> <p>You can administer user and group accounts from the YaST Control Center by selecting Security and Users > User Management and Security and Users > Group Management, respectively.</p>

Objective	Summary
4. Find the YaST Module You Need	<p>YaST offers modules that cover the functionality of the RHEL system-config-* tools. In addition to these modules there are many more YaST modules for various system administration tasks.</p> <p>yast -l lists the available modules.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 3 Network Configuration

The `system-config-network*` tools are used on RHEL to configure the network. The corresponding tool box on SUSE Linux Enterprise Server 10 is available in the Network Devices section of YaST.

Network configuration from the command line is possible in both operating systems with more or less identical commands.

The files used to store the network configuration differ between RHEL and SLES 10; this section covers the differences.

Objectives

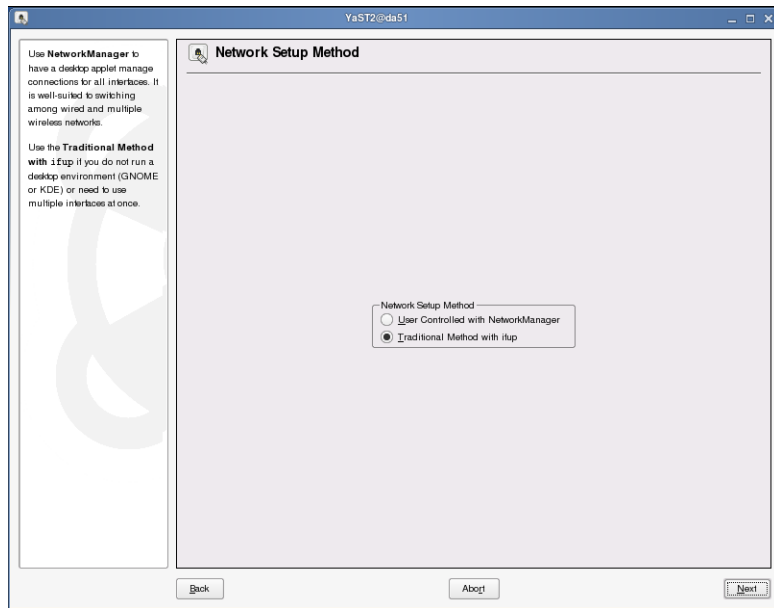
1. Manage the Network with YaST
2. Configure the Network Manually
3. Configure Host Name and Name Resolution
4. Use the NetworkManager to Configure the Network

Objective 1 Manage the Network with YaST

The YaST module for configuring network cards and the network connection can be accessed from the YaST Control Center.

To activate the network configuration module, select **Network Devices > Network Card**.

Figure 3-1

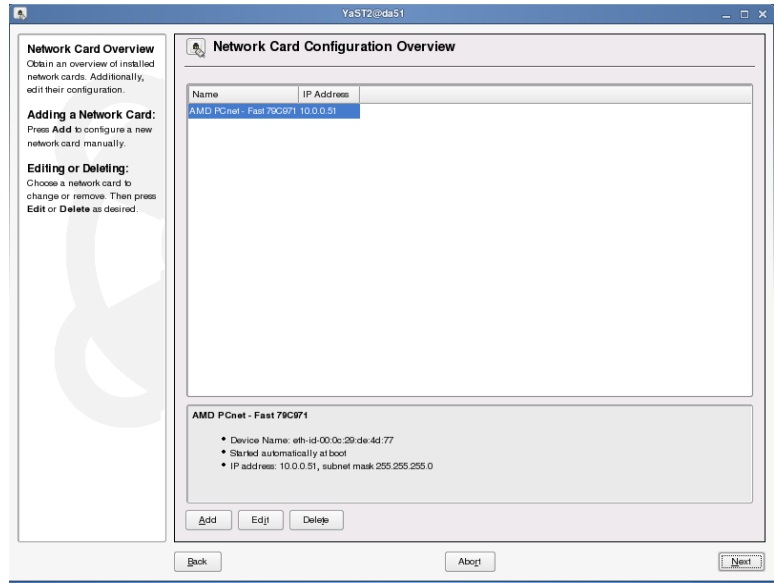


YaST wants to know the network setup method:

- **User Controlled with NetworkManager.** Use a desktop applet that manages the connections for all network interfaces. (This is mainly useful on mobile computers.)
- **Traditional Method with ifup.** The traditional method uses the command **ifup**. (We recommend to use this setup method on a server.)

Using the traditional method the next dialog shows the detected network cards.

Figure 3-2

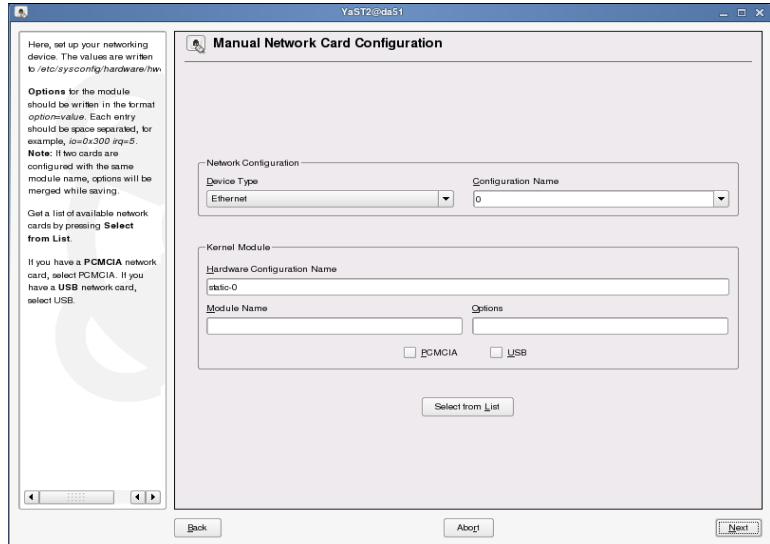


Select the card you want to configure; then select **Edit**.

Usually the cards are autodetected by YaST, and the correct kernel module is used.

If the card is not recognized by YaST, the required module must be entered manually in YaST. Select **Add**. A Manual Card Setup dialog appears:

Figure 3-3

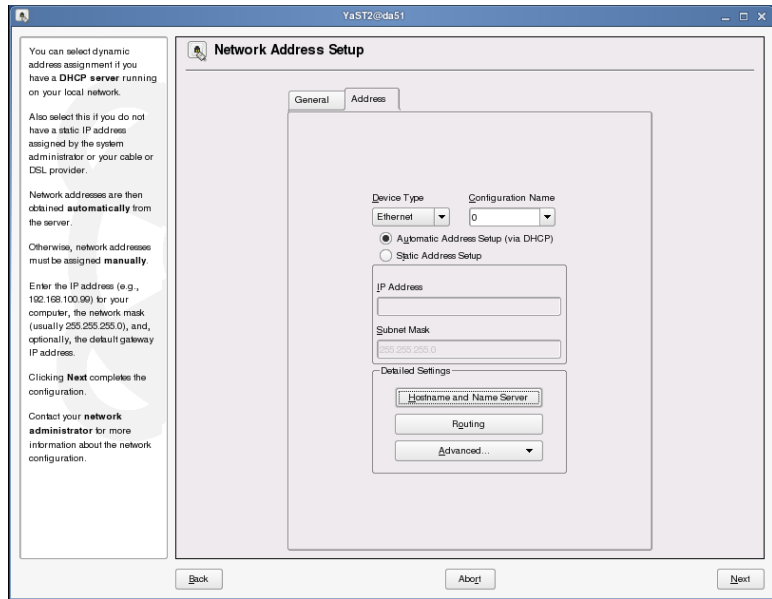


From this dialog, you enter details of the interface to configure such as Network Device Type (**Ethernet**) and Configuration Name (**0**). Under **Kernel Module**, enter the name of the module to load. You can select the card model from a list of network cards.

Some kernel modules can be configured more precisely by adding options or parameters for the kernel. Details about parameters for specific modules can be found in the kernel documentation.

After selecting **Next**, the following dialog appears:

Figure 3-4



From this dialog you enter the following information to integrate the network device into an existing network:

- **Automatic address setup (via DHCP).** Select this option if the network card should receive an IP address from a DHCP server.
- **Static address setup.** If you choose this option, you need to enter the IP address of the network interface or of the computer in the network under **IP Address**.

Each computer in the network has at least one address for each network interface, which must be unique in the entire network. According to the currently valid standard (IPv4), this address consists of a sequence of four bytes, separated by dots (such as 10.10.0.69).

When choosing the IP address, you need to know if the computer will be directly connected to the Internet. In this case, use an assigned official IP address. Otherwise, use an address from a private address space.

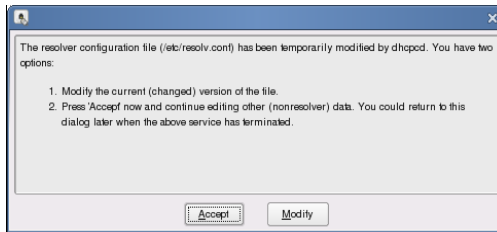
- **Subnet Mask.** The network mask (referred to as subnet mask in YaST), determines in which network an IP address is located.

The mask divides the IP address into a network section and a host section, thus defining the size of a network. All computers within the network can reach each other directly without a router in between.

- **Hostname and Name Server.** Computers in the network can be addressed directly using their IP addresses or with a unique name. A name server (DNS) must exist for the resolution of names into IP addresses and vice versa.

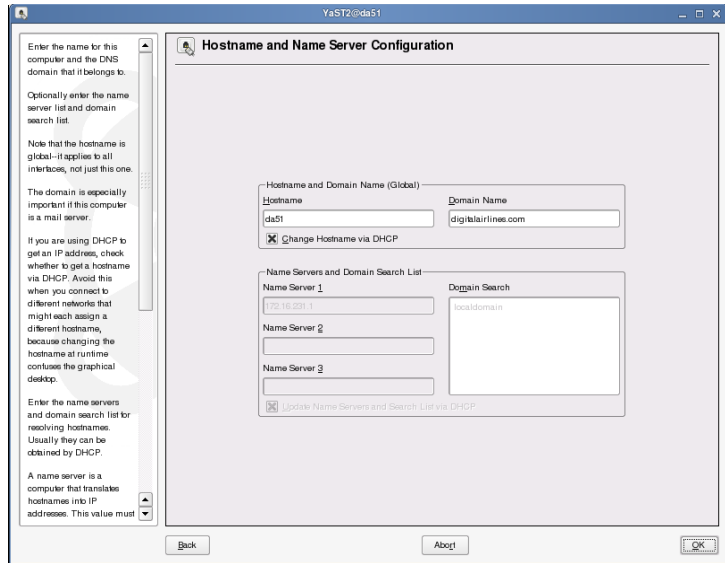
After selecting **Hostname and Name Server** and if you are using DHCP, the following appears:

Figure 3-5



If you want to change data delivered by DHCP (e.g., IP number), select **Modify**. If you only want to change other information you can select **Accept** here.

Figure 3-6



This dialog lets you enter the following:

- ❑ **Hostname.** Enter a name with which the computer can be addressed. This name should be unique within the network.
- ❑ **Domain Name.** This is the name of the DNS domain to which the computer belongs. Domains help to divide networks. All computers in a defined organizational area normally belong to the same domain.

A computer can be addressed uniquely by giving its FQDN (Fully Qualified Domain Name). This consists of the host name and the name of the domain, such as **da51.digitalairlines.com**. In this case, the domain would be digitalairlines.com.

- **List of name servers.** To address other computers in the network with their host names, identify the name server, which guarantees the conversion of computer names to IP addresses and vice versa.

You can specify a maximum of three name servers.

- **Domain search list.** In the local network, it is more appropriate to address other hosts not with their FQDN, but with their host names. The domain search list specifies the domains with which the system can expand the host name to the FQDN.

This complete name is then passed to the name server to be resolved. For example, **da51** is expanded with the search list **digitalairlines.com** to the FQDN

da51.digitalairlines.com. This name is then passed to the name server to be resolved.

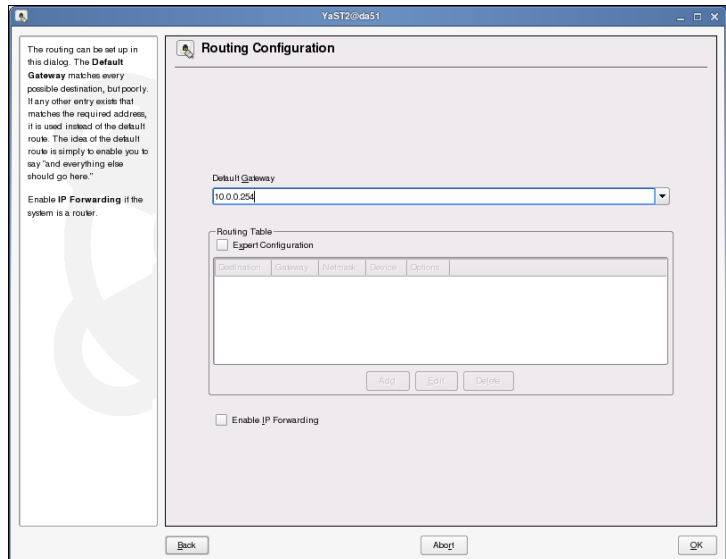
If the search list contains several domains, the completion takes place one after the other, and the resulting FQDN is passed to the name server until an entry returns an associated IP address.

Separate the domains with commas or white space.

- **Routing.** If the computer is intended only to reach other computers in the same subnet, then it is not necessary to enter any routes.

However, if you need to enter a default gateway or create a routing table, select **Routing** from the Network address setup dialog. The following appears:

Figure 3-7



You can define the following:

- ❑ **Default Gateway.** If the network has a gateway (a computer that forwards information from a network to other networks), its address can be specified in the network configuration.

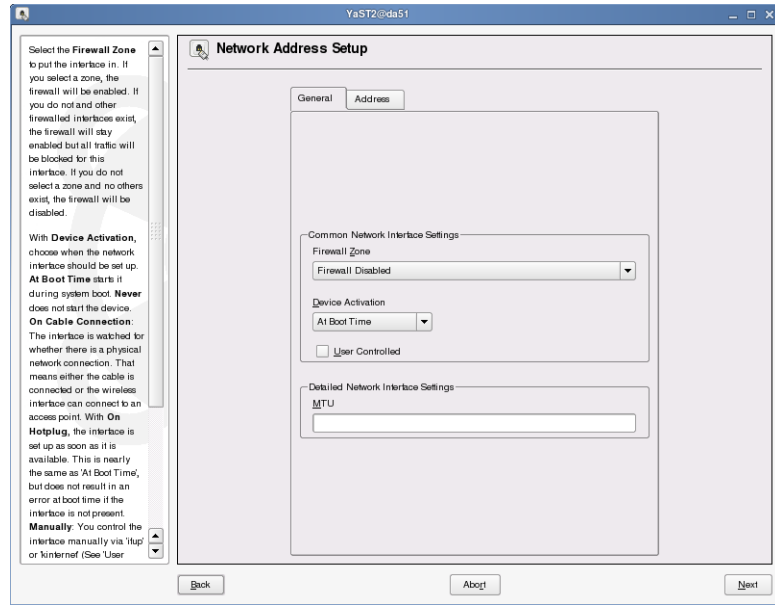
All data not addressed to the local network is then forwarded directly to the gateway.

- ❑ **Routing Table.** You can create entries in the routing table of the system after selecting **Expert Configuration**.
- ❑ **Enable IP Forwarding.** If you select this option IP packages that are not dedicated for your computer are routed.

All the necessary information is now available to activate the network card.

In the General tab of the Network Address Setup dialog, you can set up a few more options.

Figure 3-8



- **Firewall Zone.** (De-)activate the firewall for the interface. If activated, you can specify the zone to put the interface in. Three zones are possible:
 - ❑ **Internal Zone**
 - ❑ **Demilitarized Zone**
 - ❑ **External Zone**
- **Device Activation.** Choose when the interface should be set up. Possible values are:
 - ❑ **At Boot Time.** During system start

- ❑ **On Cable Connection.** If there is a physical network connection.
- ❑ **On Hotplug.** When the hardware is plugged in.
- ❑ **Manually.**
- ❑ **Never.**

Normally only root is allowed to activate and deactivate a network interface. To allow this for normal users activate the option **User Controlled**.

- **MTU.** (Maximum Transfer Unit) Maximum size of an IP package. The size depends on the hardware (Ethernet: max. 1,500 Bytes).

After you save the configuration with YaST, the ethernet card should be available in the computer. You can verify this with the command **ip**, as in the following:

```
da51:~ # ip address show
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
    inet6 ::1/128 scope host
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast
    qlen 100
    link/ether 00:e0:7d:9e:02:e8 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.51/24 brd 10.0.0.255 scope global eth0
    inet6 fec0::1:200:1cff:feb5:6516/64 scope site dynamic
    valid_lft 2591994sec preferred_lft 604794sec
    inet6 fe80::200:1cff:feb5:6516/10 scope link
3: sit0@NONE: <NOARP> mtu 1480 qdisc noop
    link/sit 0.0.0.0 brd 0.0.0.0
```

In this example, the interface eth0 was configured.

Two network devices are always set up by default-the loopback device (lo) and the device sit0@NONE, which is needed for integrating cards in networks with IPv6.

If you run this command as a user other than root, you must enter the absolute path to the command (**/sbin/ip**).

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Exercise 3-1 *Manage the Network Configuration Information from YaST*

Up to now, your system got all network configuration information via DHCP. In this exercise you change all the important information into static values.

You will find this exercise in the workbook.

(End of Exercise)

Objective 2 **Configure the Network Manually**

Although almost every step of a network configuration is done for you when you use YaST, it's sometimes useful to configure the network settings manually. For testing and troubleshooting, it can be much faster to change the network setup from the command line.

In this section, you learn how to configure network devices manually. You also learn how to configure routing with command line tools and how to save the network setup to configuration files.

- Set Up Network Interfaces with the `ip` Tool
- Set Up Routing with the `ip` Tool

Set Up Network Interfaces with the `ip` Tool

You normally configure a network card with YaST during or after installation. You can use the tool **ip** to change the network interface configuration quickly from the command line.

The command `ip` is available on RHEL as well as on SUSE Linux Enterprise Server 10.

Changing the network interface configuration at the command line is especially useful for testing purposes; if you want a configuration to be permanent, you must save it in a configuration file. These configuration files are generated automatically when you set up a network card with YaST.

You can use `ip` to perform the following tasks:

- Display the Current Network Configuration
- Change the Current Network Configuration



You can enter **/sbin/ip** as a normal user to display the current network setup only. To change the network setup, you have to be logged in as root.

As changes made with `ip` are lost with the next reboot, you also have to know how to:

- Save Device Settings to a Configuration File

Display the Current Network Configuration

With the `ip` tool, you can display the following information:

- IP Address Setup
- Device Attributes
- Device Statistics

IP Address Setup

To display the IP address setup of all interfaces, enter **ip address show**. Depending on your network setup, you see information similar to the following:

```
da2:~ # ip address show
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,NOTRAILERS,UP> mtu 1500 qdisc
pfifo_fast qlen 1000
    link/ether 00:30:05:4b:98:85 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.2/24 brd 10.0.0.255 scope global eth0
    inet6 fe80::230:5ff:fe4b:9885/64 scope link
        valid_lft forever preferred_lft forever
3: sit0: <NOARP> mtu 1480 qdisc noqueue
    link/sit 0.0.0.0 brd 0.0.0.0
```

The information is grouped by network interfaces. Every interface entry starts with a digit, called the *interface index*, with the interface name displayed after the interface index.

In the above example, there are 3 interfaces:

- **lo.** The loopback device, which is available on every Linux system, even when no network adapter is installed. (As stated above, “device” and “interface” are often used synonymously in the context of network configuration.) Using this virtual device, applications on the same machine can use the network to communicate with each other.

For example, you can use the IP address of the loopback device to access a locally installed web server by typing **http://127.0.0.1** in the address bar of your web browser.

- **eth0.** The first Ethernet adapter of the computer in this example. Ethernet devices are normally called eth0, eth1, eth2, and so on.
- **sit0.** This is a special virtual device which can be used to encapsulate IPv4 into IPv6 packets. It’s not used in a normal IPv4 network.

You always have the entries for the loopback and sit devices. Depending on your hardware setup, you might have more Ethernet devices in the ip output.

Several lines of information are displayed for every network interface, such as eth0 in the preceding example:

```
2: eth0: <BROADCAST,MULTICAST,NOTRAILERS,UP> mtu 1500 qdisc
pfifo_fast qlen 1000
```

The most important information of the line in this example is the interface index (2) and the interface name (**eth0**).

The other information shows additional attributes set for this device, such as the hardware address of the Ethernet adapter (00:30:05:4b:98:85):

```
link/ether 00:30:05:4b:98:85 brd ff:ff:ff:ff:ff:ff
```

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

In the following line, the IPv4 setup of the device is displayed:

```
inet 10.0.0.2/24 brd 10.0.0.255 scope global eth0
```

The IP address (**10.0.0.2**) follows `inet`, and the broadcast address (**10.0.0.255**) after `brd`. The length of the network mask is displayed after the IP address, separated by a `/`. The length is displayed in bits (**24**).

The following lines show the IPv6 configuration of the device:

```
inet6 fe80::230:5ff:fe4b:9885/64 scope link  
valid_lft forever preferred_lft forever
```

The address shown here is automatically assigned, even though IPv6 is not used in the network that is connected with the device. The address is generated from the hardware address of the device.

Depending on the device type, the information can differ. However, the most important information (such as assigned IP addresses) is always shown.

Device Attributes

If you are only interested in the device attributes and not in the IP address setup, you can enter **ip link show**:

```
da2:~ # ip link show  
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
2: eth0: <BROADCAST,MULTICAST,NOTRAILERS,UP> mtu 1500 qdisc  
pfifo_fast qlen 1000  
    link/ether 00:30:05:4b:98:85 brd ff:ff:ff:ff:ff:ff  
3: sit0: <NOARP> mtu 1480 qdisc noqueue  
    link/sit 0.0.0.0 brd 0.0.0.0
```

The information is similar to what you have seen when entering **ip address show**, but the information about the address setup is missing. The device attributes are displayed in brackets right after the device name.

The following is a list of possible attributes and their meanings:

- **UP.** The device is turned on. It is ready to accept packets for transmission and it's ready to receive packets from the network.
- **LOOPBACK.** The device is a loopback device.
- **BROADCAST.** The device can send packets to all hosts sharing the same network.
- **POINTOPOINT.** The device is only connected to one other device. All packets are sent to and received from the other device.
- **MULTICAST.** The device can send packets to a group of other systems at the same time.
- **PROMISC.** The device listens to all packets on the network, not only to those sent to the device's hardware address. This is usually used for network monitoring.

Device Statistics

You can use the option **-s** with the command **ip** to display additional statistics information about the devices. The command looks like the following:

ip -s link show eth0

By giving the device name at the end of the command line, the output is limited to one specific device. This can also be used to display the address setup or the device attributes.

The following is an example of the information displayed for the device eth0:

```
da2:~ # ip -s link show eth0
2: eth0: <BROADCAST,MULTICAST,NOTRAILERS,UP> mtu 1500 qdisc
pfifo_fast qlen 1000
    link/ether 00:30:05:4b:98:85 brd ff:ff:ff:ff:ff:ff
    RX: bytes    packets  errors  dropped overrun mcast
    849172787    9304150    0        0        0        0
    TX: bytes    packets  errors  dropped carrier collsns
    875278145    1125639    0        0        0        0
```

Two additional sections with information are displayed for every device. Each of the sections has a headline with a description of the displayed information.

The section starting with RX displays information about received packets, and the section starting with TX displays information about sent packets.

The sections display the following information:

- **Bytes.** The total number of bytes received or transmitted by the device.
- **Packets.** The total number of packets received or transmitted by the device.
- **Errors.** The total number of receiver or transmitter errors.
- **Dropped.** The total number of packets dropped due to a lack of resources.
- **Overrun.** The total number of receiver overruns resulting in dropped packets.

As a rule, if a device is overrun, it means that there are serious problems in the Linux kernel or that your computer is too slow for the device.

- **Mcast.** The total number of received multicast packets. This option is supported by only a few devices.

- **Carrier.** The total number of link media failures, because of a lost carrier.
- **Collsns.** The total number of collision events on Ethernet-like media.
- **Compressed.** The total number of compressed packets.

Change the Current Network Configuration

You can also use the `ip` tool to change the network configuration by performing the following tasks:

- Assign an IP Address to a Device
- Delete the IP Address from a Device
- Change Device Attributes

Assign an IP Address to a Device

To assign an address to a device, use a command similar to the following:

```
da2:~ # ip address add 10.0.0.2/24 brd + dev eth0
```

In this example, the command assigns the IP address **10.0.0.2** to the device **eth0**. The network mask is **24** bits long, as determined by the **/24** after the IP address. The **brd +** option sets the broadcast address automatically as determined by the network mask.

You can enter **ip address show dev eth0** to verify the assigned IP address. The assigned IP address is displayed in the output of the command line.

You can assign more than one IP address to a device.

Delete the IP Address from a Device

To delete the IP address from a device, use a command similar to the following:

```
da2:~ # ip address del 10.0.0.2 dev eth0
```

In this example, the command deletes the IP address **10.0.0.2** from the device **eth0**.

Use **ip address show eth0** to verify that the address was deleted.

Change Device Attributes

You can also change device attributes with the ip tool. The following is the basic command to set device attributes:

ip link set *device attribute*

The possible attributes are described in “Device Attributes” on page 3-17. The most important attributes are *up* and *down*. By setting these attributes, you can enable or disable a network device.

To enable a network device (such as eth0), enter the following command:

```
da2:~ # ip link set eth0 up
```

To disable a network device (such as eth0), enter the following command:

```
da2:~ # ip link set eth0 down
```

Save Device Settings to a Configuration File

All device configuration changes you make with **ip** are lost when the system is rebooted. To restore the device configuration automatically when the system is started, the settings need to be saved in configuration files.

The configuration files for network devices are located in the directory **/etc/sysconfig/network/**.

If the network devices are set up with YaST, one configuration file is created for every device.

For Ethernet devices, the filenames consist of ifcfg-eth-id- and the hardware address of the device. For a device with the hardware address 00:30:05:4b:98:85, the filename would be ifcfg-eth-id-00:30:05:4b:98:85.

We recommend that you set up a device with YaST first and make changes in the configuration file. Setting up a device from scratch is a complex task, because the hardware driver also needs to be configured manually.

If you have more than one network adapter in your system, it might be difficult to find the corresponding configuration file for a device.

You can use the command **ip link show** to display the hardware address for each Ethernet device. Because the hardware address is part of the file name, you can identify the right configuration file.

The content of the configuration files depends on the configuration of the device. To change the configuration file, you need to know how to do the following:

- Configure a Device Staticly
- Configure a Device Dynamically with DHCP
- Start and Stop Configured Interfaces

Configure a Device Statically

The content of a configuration file of a statically configured device is similar to the following:

```
BOOTPROTO='static'
BROADCAST=''
ETHTOOL_OPTIONS=''
IPADDR='10.0.0.2'
MTU=''
NAME='Digital DECchip 21142/43'
NETMASK='255.255.255.0'
NETWORK=''
REMOTE_IPADDR=''
STARTMODE='auto'
UNIQUE='rBUF.+xOL8ZCSAQC'
USERCONTROL='no'
_nm_name='bus-pci-0000:00:0b.0'
ETHTOOL_OPTIONS=''
```

The configuration file includes several lines. Each line has an option and a value assigned to that option, as explained below:

■ **BOOTPROTO='static'**

The option **BOOTPROTO** determines the way the device is configured. There are 2 possible values:

- ❑ **Static.** The device is configured with a static IP address.
- ❑ **DHCP.** The device is configured automatically with a DHCP server.

■ **REMOTE_IPADDR=''**

You need to set the value for the **REMOTE_IPADDR** option only if you are setting up a point-to-point connection.

■ **STARTMODE='onboot'**

The **STARTMODE** option determines how the device is started. The option can include the following values:

- ❑ **auto.** The device is started at boot time or when initialized at runtime.

- **manual.** The device must be started manually with ifup.
 - **ifplugd.** The interface is controlled by ifplugd. If you want to use interfaces mutually exclusive, also set IFPLUGD_PRIORITY
- **UNIQUE='rBUF.+xOL8ZCSAQC'**
_nm_name='bus-pci-0000:00:0b.0'

These 2 lines contain options added by YaST when the device is configured. They don't affect the network configuration itself.

- **BROADCAST=""**
IPADDR='10.0.0.2'
NETMASK='255.255.255.0'
NETWORK=""

These 4 lines contain the options for the network address configuration. The options have the following meanings:

- **BROADCAST.** The broadcast address of the network. If empty, the broadcast address is derived from the IP address and the netmask, according to the configuration in */etc/sysconfig/network/config*.
 - **IPADDR.** The IP address of the device.
 - **NETMASK.** The network mask.
 - **NETWORK.** The address of the network itself.
- **MTU=""**

You can use the MTU option to specify a value for the MTU (Maximum Transmission Unit). If you don't specify a value, the default value is used. For an Ethernet device, the default value is 1500 bytes.

- **ETHTOOL_OPTIONS=""**

ethtool is used for querying settings of an Ethernet device and changing them, for instance setting the speed or half/full duplex mode. The manual page for ethtool lists the available options.

If you want `ethtool` to modify any settings, list the options here; if no options are listed, `ethtool` is not called.

The file `/etc/sysconfig/network/ifcfg.template` contains a template that you can use as a base for device configuration files. It also has comments explaining the various options.

Configure a Device Dynamically with DHCP

If you want to configure a device by using a DHCP server, you set the `BOOTPROTO` option to **dhcp** as shown in the following:

BOOTPROTO='dhcp'

When the device is configured by using DHCP, you don't need to set any options for the network address configuration in the file. If there are any settings, they are overwritten by the settings of the DHCP server.

Start and Stop Configured Interfaces

To apply changes to a configuration file, you need to stop and restart the corresponding interface. You can do this with the commands `ifdown` and `ifup`.

For example, entering **`ifdown eth0`** disables the device `eth0`. **`ifup eth0`** enables `eth0` again.

When the device is restarted, the new configuration is read from the configuration file.



Configuring the interfaces with IP addresses, routes, etc. with the `ip` tool requires an existing device setup, including a correctly loaded kernel module. This is usually done at boot time by `/sbin/hwup`, using the configuration contained in files in the directory `/etc/sysconfig/hardware/`. Information is available in the manual page for `hwup`.



Under certain circumstances physical network devices can change the interface name, for instance the interface that used to be called eth0 now becomes eth1 and vice versa. Sometimes this happens from one boot to the next, even without any physical changes on the hardware. Information on how to achieve persistent interface names is contained in the file `/usr/share/doc/packages/sysconfig/README.Persistent_Interface_Names`.

Set Up Routing with the ip Tool

You can use the ip tool to configure the routing table of the Linux kernel. The routing table determines the path IP packets use to reach the destination system.



Because routing is a very complex topic, this objective only covers the most common routing scenarios.

You can use the ip tool to perform the following tasks:

- View the Routing Table
- Add Routes to the Routing Table
- Delete Routes from the Routing Table

As changes made with ip are lost with the next reboot, you also have to know how to:

- Save Routing Settings to a Configuration File

View the Routing Table

To view the current routing table, enter **ip route show**. For most systems, the output looks similar to the following:

```
da2:~ # ip route show
10.0.0.0/24 dev eth0 proto kernel scope link src 10.0.0.2
169.254.0.0/16 dev eth0 scope link
127.0.0.0/8 dev lo scope link
default via 10.0.0.254 dev eth0
```

Every line represents an entry in the routing table. Each line in the example is shown and explained below:

- **10.0.0.0/24 dev eth0 proto kernel scope link src 10.0.0.2**

This line represents the route for the local network. All network packets to a system in the same network are sent directly through the device **eth0**.

- **169.254.0.0/16 dev eth0 scope link**

This line shows a network route for the **169.254.0.0** network. Hosts can use this network for address auto configuration.

SLES 10 automatically assigns a free IP address from this network when no other device configuration is present. The route to this network is always set, especially when the system itself has no assigned IP address from that network

- **127.0.0.0/8 dev lo scope link**

This is the route for the loopback device.

- **default via 10.0.0.254 dev eth0**

This line is the entry for the **default** route. All network packets that cannot be sent according to the previous entries of the routing table are sent through the gateway defined in this entry.

Depending on the setup of your machine, the content of the routing table varies. In most cases, you have at least 2 entries in the routing table:

- One route to the local network the system is connected to
- One route to the default gateway for all other packets

Add Routes to the Routing Table

The following are the most common tasks you do when adding a route:

- Set a Route to the Locally Connected Network
- Set a Route to a Different Network
- Set a Default Route



Remember to substitute your own network and gateway addresses when using the following examples in a production environment.

Set a Route to the Locally Connected Network

The following command sets a route to the locally connected network:

```
da2:~ # ip route add 10.0.0.0/24 dev eth0
```

This system in this example is in the **10.0.0.0** network. The network mask is **24** bits long (255.255.255.0). All packets to the local network are sent directly through the device **eth0**.

Set a Route to a Different Network

The following command sets a route to different network:

```
da2:~ # ip route add 192.168.1.0/24 via 10.0.0.100
```

All packets for the network **192.168.1.0** are sent through the gateway **10.0.0.100**.

Set a Default Route

The following command sets a default route:

```
da2:~ # ip route add default via 10.0.0.254
```

Packets that cannot be sent according to previous entries in the routing table are sent through the gateway **10.0.0.254**.

Delete Routes from the Routing Table

To delete an entry from the routing table, use a command similar to the following:

```
da2:~ # ip route delete 192.168.1.0/24 dev eth0
```

This command deletes the route to the network **192.168.1.0** assigned to the device **eth0**.

Save Routing Settings to a Configuration File

Routing settings made with the ip tool are lost when you reboot your system. Settings have to be written to configuration files to be restored at boot time.

Routes to the directly connected network are automatically set up when a device is started. All other routes are saved in the configuration file `/etc/sysconfig/network/routes`.

The following shows the content of a typical configuration file:

```
192.168.1.0 10.0.0.100 255.255.255.0 eth-id-00:30:05:4b:98:85
default 10.0.0.254 - -
```

Each line of the configuration file represents an entry in the routing table. Each line is shown and explained below:

- **192.168.1.0 10.0.0.100 255.255.255.0
eth-id-00:30:05:4b:98:85**

All packets sent to the network **192.168.1.0** with the network mask **255.255.255.0** are sent to the gateway **10.0.0.100** through the device with the id **eth-id-00:30:05:4b:98:85**. The id is the same as used for the device configuration file.

- **Default 10.0.0.254 - -**

This entry represents a default route. All packets that are not affected by the previous entries of the routing table are sent to the gateway **10.0.0.254**. It's not necessary to fill out the last 2 columns of the line for a default route.

To apply changes to the routing configuration file, you need to restart the affected network device with the commands **ifdown** and **ifup**.

Exercise 3-2 *Configure the Network Connection Manually*

In this exercise, you learn how to configure the network manually.

You will find this exercise in the workbook.

(End of Exercise)

Objective 3 **Configure Host Name and Name Resolution**

The host name and the name resolution can also be set up manually. In this objective, you learn how to do the following:

- Set the Host and Domain Name
- Configure Name Resolution
- Files Holding the Network Configuration

Set the Host and Domain Name

The host name is configured in the file `/etc/HOSTNAME`.

The content of the file is similar to the following:

```
da2.digitalairlines.com
```

The file contains the fully qualified domain name of the system, in this case, **da2.digitalairlines.com**.

Configure Name Resolution

The name resolution is configured in the file `/etc/resolv.conf`.

The content of the file is similar to the following:

```
search digitalairlines.com
nameserver 10.0.0.254
nameserver 10.10.0.1
nameserver 10.0.10.1
```

The file contains 2 types of entries:

- **search.** The domain name in this option is used to complete incomplete host names. For example, if you look up the host name da3, the name is automatically completed to the fully qualified domain name da3.digitalairlines.com.
- **nameserver.** Every entry starting with nameserver is followed by an IP address of a name server. You can configure up to 3 name servers. If the first name server fails, the next one is used.

Files Holding the Network Configuration

The following table lists the files used to store the network configuration on RHEL and SUSE Linux Enterprise Server 10. The content of the files is not exactly identical, but the table should nevertheless help you to find the information you are looking for.

Table 3-1

RHEL	SLES 10
/etc/sysconfig/networking/devices/ ifcfg-ethx, hard-linked to /etc/sysconfig/networking/profiles/ default/ifcfg-ethx, and to /etc/sysconfig/network-scripts/ ifcfg-ethx.	/etc/sysconfig/network/ifcfg-eth-id- macaddress
/etc/sysconfig/networking/profiles/ default/hosts, hard-linked to /etc/hosts	/etc/hosts
/etc/sysconfig/networking/profiles/ default/resolv.conf, hard-linked to /etc/resolv.conf	/etc/resolv.conf
/etc/nsswitch.conf	/etc/nsswitch.conf
/etc/sysconfig/network	/etc/HOSTNAME
/etc/sysconfig/network-scripts/	/etc/sysconfig/network/scripts/

Table 3-1

RHEL	SLES 10
/etc/sysconfig/networking/devices/ route-ethx, hard-linked to /etc/sysconfig/networking/profiles/ default/route-ethx, and to /etc/sysconfig/network-scripts/ route-ethx.	/etc/sysconfig/network/routes

Objective 4 **Use the NetworkManager to Configure the Network**

In case you are using SUSE Linux Enterprise Server 10 on a laptop, you will most likely use different kinds of Internet access, depending on where you are—maybe a LAN in your office and a wireless connection at a customer site.

The conventional network setup requires you to switch to the root account to change the network configuration. The purpose of the **NetworkManager** (package `NetworkManager`) is to allow the user to change the network configuration according to his needs, without switching to the root account.

NetworkManager runs as a root-user system level daemon, since root privileges are needed to manipulate hardware directly. The programs used for this purpose are `/usr/sbin/NetworkManager` and `/usr/sbin/NetworkManagerDispatcher`. **nm-tools** can be used to list information about NetworkManager, devices, and wireless networks.

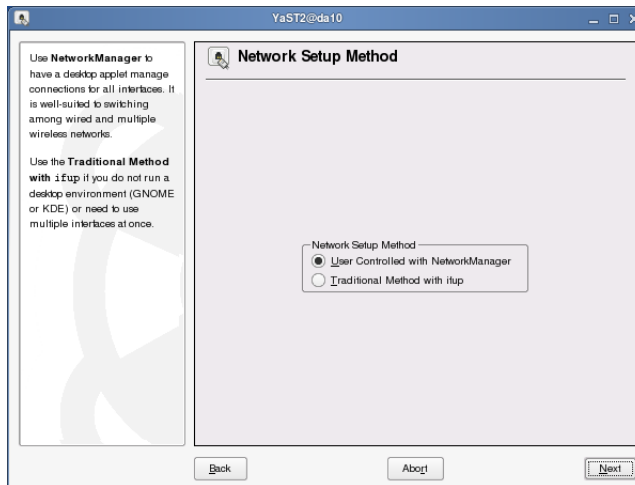
From a list of all adapters currently installed on the system, NetworkManager will first try a wired and then a wireless adapter. Wireless adapters that support wireless scanning are preferred over ones that cannot. NetworkManager does not try to keep a connection up as long as possible, meaning that plugging into a wired network will switch the connection to the wired network, away from the wireless one.

For wireless networking support, NetworkManager keeps two lists of wireless networks: a **Trusted list**, and a **Preferred list**. The trusted list contains networks the user specifically adds to it, while the preferred list contains networks the user forces NetworkManager to connect to.

Since trusted and preferred networks are user-specific, there must be some mechanism of getting and storing this information per user. This is achieved with a desktop-level per-user process, **nm-applet**, or KNetworkManager in KDE. NetworkManager communicates over DBUS with these user level processes.

Switching to NetworkManager is done by starting YaST and selecting **Network Devices > Network Cards**. In the **Network Setup Method** dialog, you select **User Controlled with NetworkManager**:

Figure 3-9

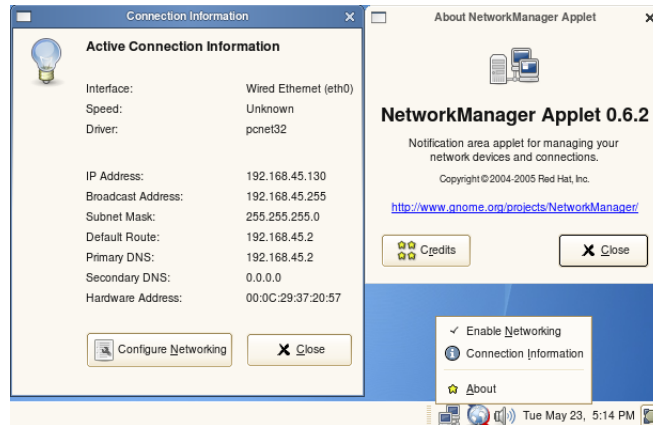


The following dialogs of this module are the same for both setup methods.

When selecting **User Controlled with NetworkManager**, YaST sets the variable **NETWORKMANAGER=** in `/etc/sysconfig/network/config` to “yes”.

Choosing the NetworkManager in YaST will also automatically start the Network Applet when a user logs in. Using the desktop applet, the user can easily change the network configuration:

Figure 3-10



Note: As there was no wireless card built into the computer on which the above screenshot was taken, there is no option for switching networks in this screenshot.

Summary

Objective	Summary
1. Manage the Network with YaST	<p>The YaST module for configuring the network card and the network connection can be found at Network Devices > Network Card.</p> <p>The following details are needed to integrate the network device into an existing network:</p> <ul style="list-style-type: none">■ Method of network setup■ Static IP address■ Network mask■ Host name■ Name server■ Routing (gateway)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
2. Configure the Network Manually	<p>You can perform the following tasks with the ip tool:</p> <ul style="list-style-type: none">■ Display the IP address setup: ip address show■ Display device attributes: ip link show■ Display device statistics: ip -s link show■ Assign an IP address: ip address add <i>IP_address/netmask brd + dev device_name</i>■ Delete an IP address: ip address del <i>IP_address dev device_name</i> <p>On SLES 10, the configuration files for network devices are located in /etc/sysconfig/network/.</p> <p>Configured devices can be enabled with ifup <i>device_name</i> and disabled with ifdown <i>device_name</i>.</p>

Objective	Summary
. Set Up Routing with the ip Tool	<p>You can perform the following tasks with the ip tool:</p> <ul style="list-style-type: none"> ■ View the routing table: ip route show ■ Add routes to the routing table ip route add <i>network/netmask</i> dev <i>device_name</i> ■ Delete routes from the routing table ip route del <i>network/netmask</i> dev <i>device_name</i> <p>The configuration for the routing table is located in the file <code>/etc/sysconfig/network/routes</code>.</p>
3. Configure Host Name and Name Resolution	<p>The host name is configured in the file <code>/etc/HOSTNAME</code>.</p> <p>The name resolution is configured in the file <code>/etc/resolv.conf</code>.</p> <p>One line specifies the search domain; the others list up to three available name servers.</p>
4. Use the NetworkManager to Configure the Network	<p>NetworkManager allows the user to change the network configuration without having to assume root privileges.</p> <p>NetworkManager is mainly useful for use on laptops.</p>

SECTION 4 Manage the Linux File System

In this section, you learn how to manage your SUSE Linux Enterprise Server 10 file system by implementing partitions, creating file systems, checking the file system for errors, and setting up LVM and software RAID.

The default installation proposal for partitioning and file systems on SLES 10 differs from RHEL. The default in SLES 10 does not include LVM, and the file system suggested is ReiserFS. Unlike RHEL, SLES 10 also supports XFS.

Objectives

1. Select a Linux File System
2. Configure Linux File System Partitions
3. Manage Linux File Systems
4. Configure Logical Volume Manager (LVM) and Software RAID

Objective 1 **Select a Linux File System**

One of the key roles performed by the Linux operating system is providing storage services through creating and managing a file system.

To successfully select a file system that meets your server requirements, you need to understand the following about file systems available for Linux:

- Linux File Systems
- Linux File System Internals
- File System Journaling
- Additional File System Documentation

It is very important to keep in mind that there might be no file system that best suits all kinds of applications. Each file system has its particular strengths and weaknesses, which must be taken into account.

Always bear in mind that even the most sophisticated file system cannot be a substitute for a reasonable backup strategy.



For additional details on specific file systems (such as ext3 and ReiserFS), see Section 18.2 in the ***SLES 10 Installation and Administration manual*** (/usr/share/doc/manual/sles-admin_en/, package sles-admin_en).

Also see “Additional File System Documentation” on page 4-12 at the end of this objective.

Linux File Systems

The type of file system you select depends on several factors (including speed and journaling). While Linux supports several file systems, for a server the choice is usually between three of them.

- **ext2/ext3.** The ext2 file system is inode-based, designed for speed, is efficient, and does not fragment easily.

Because of these features, ext2 continues to be used by many administrators, even though it does not provide a journaling feature.

The ext2 file system has been available for many years, and is easily converted to an ext3 file system.

ext3. ext3 is the version of the ext2 file system that supports journaling.

- **ReiserFS.** Originally designed by Hans Reiser, ReiserFS treats the entire disk partition as if it were a single database table, storing not only the file metadata, but the file itself.

Directories, files, and file metadata are organized in an efficient data structure called a “balanced tree,” which offers significant speed improvements for many applications, especially those which use lots of small files.

- **XFS.** XFS is a high-performance journaling file system from SGI. It provides quick recovery after a crash, fast transactions, high scalability, and excellent bandwidth.

XFS combines advanced journaling technology with full 64-bit addressing and scalable structures and algorithms.



For details on XFS, see <http://oss.sgi.com/projects/xfs/>.

Linux File System Internals

File systems in Linux are characterized by the fact that data and administration information are kept separate. Each file is described by an *inode* (index node or information node).

Each of these inodes has a size of 128 bytes and contains all the information about this file except the filename. This includes details such as the owner, access permissions, the size, various time details (time of modification, last time of access, and time of modification of the inode), and the links to the data blocks of the file.

How data organization takes place differs from one file system format to the next. To understand the basics of file system data organization on Linux, you need to know the following:

- ext2fs File System Format
- ReiserFS Format
- Directories
- Network File System Formats

ext2fs File System Format

The ext2 file system format is, in many ways, identical to traditional UNIX file system formats. The concepts of inodes, blocks, and directories are the same.

When a file system is created (the equivalent of formatting in other operating systems), the maximum number of files that can be created is specified. The inode density (together with the capacity of the partition) determines how many inodes can be created.

Remember that it is not possible to generate additional inodes later. You can only specify the inode density when creating the file system.

An inode must exist for each file or directory on the partition. The number of inodes also determines the maximum possible number of files. Typically, an inode is generated for 4096 bytes of capacity.

On average, each file should be 4 KB in size for the capacity of the partition to be used optimally. If a large number of files are smaller than 4 KB, more inodes are used compared with the capacity.

This can result in the system being unable to create any more files, even if there is still space on the partition.

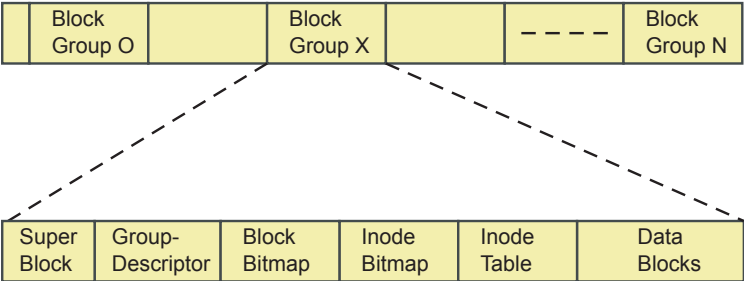
For applications that create a large number of very small files, the inode density should be increased by setting the corresponding capacity to a smaller value (such as 2048 or even 1024). However, the time needed for a file system check will increase substantially.

The space on a partition is divided into *blocks*. These have a fixed size of 1024, 2048, or 4096 bytes. You specify the block size when the file system is created; it cannot be changed later.

The block size determines how much space is reserved for a file. The larger this value is, the more space is consumed by the file, even if the actual amount of data is smaller.

In the classic file system formats (to which ext2 also belongs), data is stored in a linear chain of blocks of equal size. A specific number of blocks is grouped together in a block group (as illustrated in the following) and each block group consists of 32768 blocks:

Figure 4-1



The boot sector is located at the beginning of this chain and contains static information about the file system, including where the kernel to load can be found.

Each block group contains the following components:

- **Superblock.** The superblock is read when the file system is mounted and contains the following information about the file system:
 - The number of free and occupied blocks and inodes.
 - The number of blocks and inodes for each block.
 - Information about file system use, such as the time of the last mount, the last write access, and the number of mounts since the last file system check.
 - A valid bit, which is set to **0** when the file system is mounted and set to **1** again by umount.

When the computer is booted, the valid bit is checked. If it is set to 0 (power failure or reset), the automatic file system check is started.

The remains of files that can no longer be reconstructed are stored in the directory lost+found (in an ext2/ext3 file system).

For reasons of security, there are copies of the superblock. Because of this, the file system can be repaired, even if the first superblock has been destroyed.

- **Group Descriptor.** Information on the location of other areas (such as block bitmap and inode bitmap) is stored here. This information is stored at several locations within the file system for reasons of data security.
- **Block Bitmap.** Information is stored here indicating which blocks in this group are free or occupied.
- **Inode Bitmap.** Information is stored here indicating which inodes are free or occupied.

- **Inode Table.** File information is stored in this table that includes owners, access permissions, time stamps, and links to the data blocks in which the data is located.
- **Data Blocks.** This is where the actual data is located.

The ext2 file system format can process filenames with a length of up to 255 characters. With the path, a name can be a maximum of 4096 characters in length (slashes included).

A file can be up to 16 GB in size for a block size of 1024 bytes or two TB for a block size of 4096 bytes. The maximum file system size is two TB (with a block size of 1024 bytes) or 16 TB (with a block size of 4096 bytes).



The limitation on file size remains for the ext2 file system. However, the kernel can now handle files of almost any size.

ReiserFS Format

On a file system with ext2 and a block size of 1024 bytes, a file 8195 bytes in size occupies 8 blocks completely and a ninth block with three bytes.

Even though only three bytes are occupied, the block is no longer available. This means that approximately 11 percent of available space is wasted.

If the file is 1025 bytes in size, two blocks are required, one of which is almost completely empty. Almost 50 percent of the space is wasted.

A worst case occurs if the file is very small: even if the file is only 50 bytes in size, a whole block is used (95 percent wasted).

A solution to this problem is provided by the ReiserFS format, which organizes data in a different way. This file system format has currently a fixed block size of 4096 bytes.

However, small files are stored more efficiently. Only as much space is reserved as is actually required—not an entire block. Small files or the ends of files are stored together in the same block.

The inodes required are not generated when the file system is created, but only when they are actually needed. This allows a more flexible solution to storage requirements, increasing efficiency in the use of hard drive space.

Another advantage of the ReiserFS is that access to files is quicker. This is done through the use of balanced binary trees in the organization of data blocks.

However, balanced trees require considerably more processing power because after every file is written the entire tree must be rebalanced.

The current version of the ReiserFS (3.6) contained in the kernel since version 2.4.x allows a maximum partition size of 16 TB. A file also has a maximum size of 16 TB.

The same limitations exist for filenames as with the ext2 file system format.

Directories

Inodes contain all the administrative information for a file, but not the filename. The filename is stored in the directory.

Like a catalog, directories contain information on other files. This information includes the number of the inode for the file and its name.

Directories serve as a table in which inode numbers are assigned line-by-line to filenames. You can view the inode assigned to a filename by using the command **ls -li**, as in the following:

```
da10:~ # ls -li /
 2 .          104002 cdrom      80045 floppy   104081 mnt    103782 sbin
 2 ..         99068 dev          95657 home     81652 opt     80044 tmp
104005 bin    104004 dvd        102562 lib      1 proc       4 usr
 2 boot      95722 etc        95718 media     81598 root    80046 var
```

Each filename is preceded by the inode number.

On this particular SUSE Linux server there are 2 partitions: one holds the root directory /, and one holds the directory /boot/.

Because inodes are always uniquely defined on one partition only, the same inode numbers can exist on each partition.

In the example, the two entries “.” (a link to the current directory—here the root directory) and boot (the second partition is mounted on this directory) have the same inode number (2), but they are located on different partitions.

If you were to unmount the /boot partition, ls -li would show a different inode number, that of the directory /boot (the mountpoint) on the root partition. The same holds true for /proc.

The file “.”, which is actually a link to the previous layer in the direction of the root directory, also has an inode number of 2. Because you are already in the root directory, this link points to itself. It is another name entry for an inode number.

The table (the directory file) for the root directory can be represented as in the following example:

Table 4-1

Inode Number	Filename
2	.
2	..
4	usr
5	proc
18426	boot
80044	tmp
80045	floppy
80046	var
...	...

Network File System Formats

In addition to the already mentioned file system formats on the local computer, Linux also understands various network file system formats. The most significant of these is the Network File System (NFS), the standard in the UNIX world.

With NFS, it does not matter which file system format is used locally on individual partitions. As soon as a computer is functioning as an NFS server, it provides its file systems in a defined format NFS clients can access.

Using additional services included on SUSE Linux Enterprise Server, Linux can also work with the network file system formats of other operating systems.

These include the Server Message Block (SMB) format used in Windows and the Netware Core Protocol (NCP) from Novell.

SMB allows Linux to mount Windows 9x/NT/XP network shares.



File types, like directories, FIFOs, Sockets as well as the layout of the file system tree are covered in *SUSE Linux Enterprise Server 10 Fundamentals* (Course 3071).

File System Journaling

File systems are basically databases that store files and use file information such as the filename and timestamp (called *metadata*) to organize and locate the files on a disk.

When you modify a file, the file system performs the following transactions:

- It updates the file (the data)
- It updates the file metadata

Because there are two separate transactions, corruption can happen when only the file data is updated (but not the metadata) or vice versa, resulting in a difference between the data and metadata.

This can be caused, for instance by a power outage. The data might have been written already, but the metadata might not have been updated yet.

When there is a difference between the data and metadata, the state of the file system is inconsistent and requires a file system check and possibly repair. For ext2, this includes a walk through the entire file system, which is very time consuming on today's hard disks with hundreds of GB capacity.

In a journal-based file system, the journal keeps a record of all current transactions, and updates the journal as transactions are completed. Checking the file system, for instance after a power outage, consists mainly in replaying the journal and is much faster than checking the entire file system.

For example, when you first start copying a file from a network server to your workstation, the journaled file system submits an entry to the journal indicating that a new file on the workstation is being created.

After the file data and metadata are copied to the workstation, an entry is made indicating that the file was created successfully.

While recording entries in a journal requires extra time for creating files, it makes recovering an incomplete transaction easy as the journal can be used to repair the file system.

Additional File System Documentation

Each of the Linux file systems maintains its own home page on which to find mailing list information, further documentation, and FAQs. These include the following:

Table 4-2

File System	URL
ext2	http://e2fsprogs.sourceforge.net/
ReiserFS and Reiser4	http://www.namesys.com/
SGI's XFS	http://oss.sgi.com/projects/xfs/

A comprehensive multipart tutorial about Linux file systems can be found at IBM developerWorks at the following URL:

<http://www-106.ibm.com/developerworks/library/l-fs.html>

If you are interested in the limit of various file systems (file and file system sizes), visit http://www.novell.com/products/linuxenterpriseserver/kernel_limits.html.

The Linux Filesystem Hierarchy Standard (FHS) can be found at: <http://www.pathname.com/fhs/>

Objective 2 Configure Linux File System Partitions

A basic task of all system administrators is maintaining file system layouts. As a note of caution, you should always back up your data before working with tools that change the partition table or the file systems.

In most cases, YaST proposes a reasonable partitioning scheme during installation that can be accepted without change. However, you can also use YaST to customize partitioning after installation.

On the command line, you would first use **fdisk** to manage partitions, and then create a file system on that partition using **mkfs**.

To implement partitions on your SUSE Linux Enterprise Server, you need to know the following:

- Linux Device and Partition Names
- Design Guidelines for Implementing Partitions
- Manage Partitions with YaST
- Manage Partitions with fdisk

Linux Device and Partition Names

The following table shows the names of the Linux devices used for hard drives:

Table 4-3

Device	Linux Name
Primary master IDE hard disk	/dev/hda
Primary slave IDE hard disk	/dev/hdb
Secondary master IDE hard disk	/dev/hdc
Secondary slave IDE hard disk	/dev/hdd
First SCSI hard disk	/dev/sda

Table 4-3 *(continued)*

Device	Linux Name
Second SCSI hard disk	/dev/sdb

Partitions follow the naming convention of the device name and partition number.

For example, the first partition on the first IDE drive would be /dev/hda1 (/dev/hda + 1 as the first partition). The first logical partition defined on an IDE hard disk will always be number 5.

The following table shows the partition names corresponding to the device the partition is defined on:

Table 4-4

Partition	Linux Name
First partition on first IDE hard drive	/dev/hda1
Second partition on first IDE hard drive	/dev/hda2
First partition on third SCSI hard drive	/dev/sdc1
First logical partition on first IDE hard drive	/dev/hda5
Second logical partition on first IDE hard drive	/dev/hda6

For example, if you perform a new installation of SuSE Linux on a system with 2 IDE drives you might want the first drive to include a partition for swap and /. You might want to put all logs, mail, and home directories on the second hard drive.

The following is an example of how you might want to partition the disks (it assumes that the CD-ROM drive is the slave on the first IDE controller):

Table 4-5

Partition	Linux Name
Swap partition	/dev/hda1
/ partition	/dev/hda2

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Table 4-5 *(continued)*

Partition	Linux Name
Extended partition on second disk	/dev/hdc1
/var as a logical partition on second disk	/dev/hdc5
/home as a logical partition on second disk	/dev/hdc6
/app1 as a logical partition on second disk	/dev/hdc7



On older installations you often find a small partition for /boot/. The reason for this is that the boot loader LILO needed the kernel within the first 1024 cylinders of the hard disk to boot the machine.

Design Guidelines for Implementing Partitions

YaST normally proposes a reasonable partitioning scheme with sufficient disk space. This is usually a swap partition (between 256 and 500 MB) and with the rest of the disk space reserved for a / partition.

In addition, if there is an existing partition on the hard drive, YaST attempts to maintain that partition.

If you want to implement your own partitioning scheme, consider the following recommendations:

Disk Space Distribution

Depending on the amount of space and how the computer will be used, adjust the distribution of the available disk space. The following are some basic guidelines:

Up to 4 GB

One partition for the swap space and one root partition (/). In this case, the root partition must allow for those directories that often reside on their own partitions if more space is available.

4 GB or More

A swap partition, a root partition (1 GB), and 1 partition each for the following directories as needed:

- **/boot/**. Depending on the hardware, it might also be useful to create a boot partition (/boot) to hold the boot mechanism and the Linux kernel.

This partition should be located at the start of the disk and should be at least 20 MB or 1 cylinder.

As a rule of thumb, always create such a partition if it was included in YaST's original proposal. If you are unsure about this, create a boot partition to be on the safe side.

- **/opt/**. Some (mostly commercial) programs install their data in /opt/. In this case, you might want to create a separate partition for /opt/ (4 GB or more). For instance KDE and GNOME are installed in /opt/.
- **/usr/**. Apart from directories holding user data, /usr/ is usually the biggest directory in the Linux installation. Putting it on a separate partition allows special mount options, like read only to prevent changes to programs. Software updates require to remount the partition read-write, though.

- **/var/**. If the computer is used as a mail server, it might be a good idea to put **/var/** on a separate partition. While too much mail might still render the mail service unusable, they would just fill the partition containing the **/var** directory, not the root file system. The administrator would still be able to administer the server and correct the issue.
- **/srv/**. When the machine acts as a web or ftp server, the data offered to users could be put on a separate partition.
- **/home/**. Putting **/home/** on a separate partition prevents users from using up all disk space and facilitates updates. If you have to reinstall the operating system you can preserve data in **/home** by leaving the partition untouched.
- **/tmp/**. Having **/tmp/** on a separate partition allows you to mount it with special options, like **noexec**, and also prevents processes from filling the disk with files in **/tmp/**.
- **Additional partitions**. If the partitioning is performed by YaST and other partitions are detected in the system, these partitions are also entered in the file **/etc/fstab** to enable easy access to this data.

The following is an example:

```
dev/sda8 /data2 auto noauto,user 0 0
```

Such partitions, whether they are Linux or FAT, are specified by YaST with the options **noauto** and **user**. This allows any user to mount or unmount these partitions as needed.

For security reasons, YaST does not automatically enter the **exec** option, which is needed for executing programs from the respective location. However, you can enter this option manually.

Entering the **exec** option is necessary if you encounter system messages such as “bad interpreter” or “Permission denied”.

Manage Partitions with YaST

You can use the YaST Expert Partitioner during or after installation to customize the default or existing partition configuration.

The interface of the Expert Partitioner after installation does not differ from the interface you used during installation.

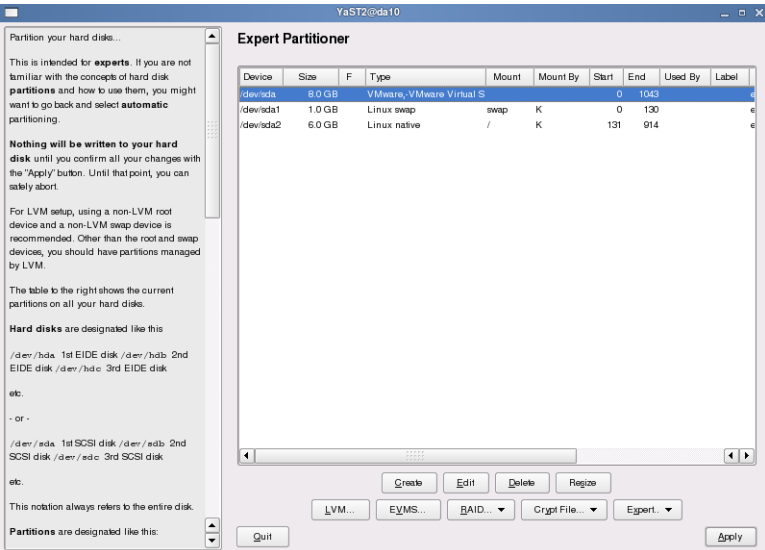
To start the Expert Partitioner, press **Alt+F2**, enter **yast2**, and enter the root password when prompted. Then select **System > Partitioner**. The following warning appears:

Figure 4-2



After selecting **Yes**, the expert partitioner appears:

Figure 4-3



The expert partitioner lets you modify the partitioning of your hard disk. You can manage the list of partitions by adding (**Create**), editing (**Edit**), deleting (**Delete**), or resizing (**Resize**) partitions.

Entire hard disks are listed as devices without numbers (such as /dev/hda or /dev/sda). Partitions are listed as parts of these devices (such as /dev/hda1 or /dev/sda1).

The size, type, file system, and mount point of the hard disks and their partitions are also displayed. The mount point describes where the partition is mounted in the Linux file system tree.

Manage Partitions with fdisk

The program **fdisk** is used for partitioning hard disks from the command line. The program **partprobe** is used to get the kernel to use the new partition table. As these tool are the same on RHEL and SLES 10 they are not covered here.

Objective 3 Manage Linux File Systems

To perform basic Linux file system management tasks in SUSE Linux Enterprise Server, you need to know how to do the following:

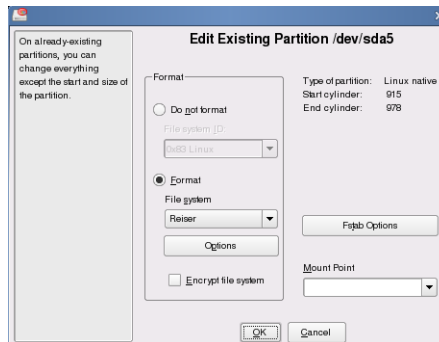
- Create a File System with YaST
- Create a File System with Command Line Tools
- Mount File Systems
- Check a File System

Create a File System with YaST

You can use YaST to create a file system (such as ext3 or ReiserFS) on a partition. This is done by starting the Expert Partitioner as root by entering in a console window **yast2 disk**. After acknowledging the warning message, the **Expert Partitioner** opens up.

To create a file system on a partition, select the partition and then select **Edit**; the following appears:

Figure 4-4



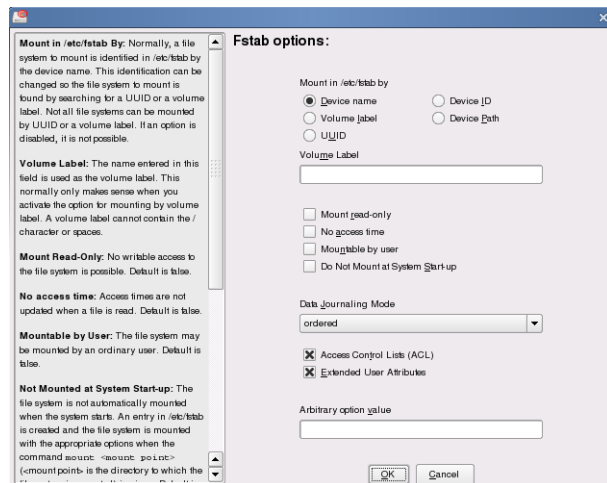
To format the partition with a file system, select **Format**. From the **File system** drop-down list, select a file system from the list of available file systems (such as **Reiser** or **Ext3**).

To view the available format options, select **Options**; the options shown depend on the file system you chose from the drop-down menu. We recommend keeping the default settings for most implementations. To return to the main format menu, select **OK**.

If you want to encrypt all data saved to the partition, select **Encrypt file system**. Encrypting a file system only prevents unauthorized mounting; once mounted the files are accessible like any other file on the system. You should only use this option for non-system partitions such as user home directories.

Select **Fstab Options** to edit the fstab entry for this partition.

Figure 4-5



These options are saved in **/etc/fstab** and are used when mounting the file system. In most cases the defaults offered don't need to be changed.

A description of each option is included in the left frame of the **Fstab options** dialog.

When you finish configuring the fstab options; select **Ok**.

In the **Mount Point** field enter the *directory* where the partition should be mounted in the file system tree. If the directory does not exist yet, it is automatically created by YaST.

When you finish configuring the file system and mounting parameters, select **OK**, and **Apply** in the Expert Partitioner dialog.

A warning message appears cautioning you about committing the changes you have made. Choosing **Apply** commits the changes to disk and returns you to the Expert Partitioner, whereas **Finish** commits them and closes the Expert Partitioner.

Create a File System with Command Line Tools

There are various commands to create file systems, including `mke2fs`, `mkfs.ext3`, and `mkreiserfs`. You can use these to create file systems, such as `ext2`, `ext3`, and `ReiserFS`.

The alternative is the command **mkfs**, which is a frontend for the actual commands that create file systems (such as `mkfs.ext2`, `mkfs.ext3`, or `mkfs.msdos`).

When using **mkfs**, you need to use the option **-t** to indicate the file system type you want to create. If you do not indicate a file system type, `mkfs` automatically creates an `ext2` file system.

You need to know how to:

- Create an `ext2` or `ext3` File System
- Create a Reiser File System

Create an ext2 or ext3 File System

Creating an ext2 or ext3 file system with `mkfs` on SUSE Linux Enterprise Server 10 does not differ from the procedure used on RHEL.

Create a Reiser File System

You can create a Reiser file system by using the command **`mkreiserfs`** or **`mkfs -t reiserfs`**:

```
da10:~ # mkfs -t reiserfs /dev/sda6
mkfs.reiserfs 3.6.19 (2003 www.namesys.com)

A pair of credits:
Yury Umanets (aka Umka) developed libreiser4, userspace plugins,
...

Guessing about desired format.. Kernel 2.6.16.14-6-smp is running.
Format 3.6 with standard journal
Count of blocks on the device: 62240
Number of blocks consumed by mkreiserfs formatting process: 8213
Blocksize: 4096
Hash function used to sort names: "r5"
Journal Size 8193 blocks (first block 18)
Journal Max transaction length 1024
inode generation number: 0
UUID: 73abdf80-2b72-4844-9967-74e99813d056
ATTENTION: YOU SHOULD REBOOT AFTER FDISK!
        ALL DATA WILL BE LOST ON '/dev/sda6'!
Continue (y/n):y
Initializing journal - 0%....20%....40%....60%....80%....100%
Syncing..ok
ReiserFS is successfully created on /dev/sda6.
```

To find out about the available options, look at **`man mkreiserfs`**. Usually there is no need to use different values than those used by default.

Mount File Systems

In Windows systems separate drive letters represent different partitions. Linux does not use letters to designate partitions, it mounts partitions to a directory in the file system. Directories used for mounting are also called *mount points*.

For example, to add a new hard disk to a Linux system, first you partition and format the drive. You then use a directory (such as `/data/`) in the file system and mount the partition to that directory using the command **mount**.

To unmount (detach) a file system, you use the **umount** command (for details, enter **man umount**).



You can also mount remote file systems, shared via the Network File System (NFS), to directories you create in your file system.

The directory `/mnt/` is used by default for temporarily mounting local and remote file systems. All removable devices are mounted by default to `/media/`, such as the following:

- A CD-ROM on `/dev/cdrom` is mounted by default to `/media/cdrom`.
- A floppy disk on `/dev/floppy` is mounted by default to `/media/floppy`.

The above is identical to the scheme used on RHEL.

When using SLES 10 from a desktop environment such as Gnome or KDE, media such as floppy disks and CDs are automatically mounted and unmounted. If the CD-ROM has a label, it is mounted to `/media/label`.

To manage mounting (and unmounting) file systems, you need to know the following:

- Configuration File for Mounting File Systems: `/etc/fstab`
- View Currently Mounted File Systems
- Mount a File System
- Unmount a File System

Configuration File for Mounting File Systems: `/etc/fstab`

The file systems and their mount points in the directory tree are configured in the file `/etc/fstab`. This file contains 1 line with 6 fields for each mounted file system.

The lines look similar to the following:

Field 1	Field 2	Field 3	Field 4	Field 5	Field 6
<code>/dev/hda2</code>	<code>/</code>	<code>reiserfs</code>	<code>acl,user_xattr</code>	<code>1</code>	<code>1</code>
<code>/dev/hda1</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>proc</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>sysfs</code>	<code>/sys</code>	<code>sysfs</code>	<code>noauto</code>	<code>0</code>	<code>0</code>
<code>debugfs</code>	<code>/sys/kernel/debug</code>	<code>debugfs</code>	<code>noauto</code>	<code>0</code>	<code>0</code>
<code>usbfs</code>	<code>/proc/bus/usb</code>	<code>usbfs</code>	<code>noauto</code>	<code>0</code>	<code>0</code>
<code>devpts</code>	<code>/dev/pts</code>	<code>devpts</code>	<code>mode=0620,gid=5</code>	<code>0</code>	<code>0</code>
<code>/dev/fd0</code>	<code>/media/floppy</code>	<code>auto</code>	<code>noauto,user, sync</code>	<code>0</code>	<code>0</code>

Each field provides the following information for mounting the file system:

- **Field 1.** Lists the name of the device file, or the file system label, or the UUID (Universally Unique Identifier). Use of **LABEL=label** or **UUID=uuid** has the advantage that the partition is mounted correctly even if the device file used changes, for instance because you swapped hard disks on the IDE controller.

- **Field 2.** Lists the mount point—the directory to which the file system should be mounted. The directory specified here must already exist. You can access the content on the media by changing to the respective directory.
- **Field 3.** Lists the file system type (such as `ext2`, `reiserfs`).
- **Field 4.** Shows the mount options. Multiple mount options are separated by commas (such as **`noauto,user,sync`**).
- **Field 5.** Indicates whether to use the backup utility **`dump`** for the file system. **0** means no backup.
- **Field 6.** Indicates the sequence of the file system checks (with the **`fsck`** utility) when the system is booted:
 - **0:** file systems that are not to be checked
 - **1:** the root directory
 - **2:** all other modifiable file systems; file systems on different drives are checked in parallel

While `/etc/fstab` lists the file systems and where they should be mounted in the directory tree during startup, it does not contain information on the actual current mounts.

The `/etc/mtab` file lists the file systems currently mounted and their mountpoints. The `mount` and `umount` commands affect the state of mounted file systems and modify the `/etc/mtab` file.

The kernel also keeps information for `/proc/mounts`, which lists all currently mounted partitions.

For troubleshooting purposes, if there is a conflict between `/proc/mounts` and `/etc/mtab` information, the `/proc/mounts` data is always more current and reliable than `/etc/mtab`.

View Currently Mounted File Systems

You can view the file systems currently mounted by entering the command **mount**. Information similar to the following appears:

```
da10:~ # mount
/dev/sda2 on / type reiserfs (rw,acl,user_xattr)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
debugfs on /sys/kernel/debug type debugfs (rw)
udev on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
securityfs on /sys/kernel/security type securityfs (rw)
```

You can also view this information in the file `/proc/mounts`.

Mount a File System

You can use the command **mount** to manually mount a file system. The general syntax for mounting a file system with mount is

**mount [-t *file_system_type*] [-o *mount_options*] *device*
*mount_point_directory***

By using mount, you can override the default settings in `/etc/fstab`.

For example, entering the following mounts the partition `/dev/hda9` to the directory `/space/`:

```
mount /dev/hda9 /space
```

You do not usually specify the file system type because it is recognized automatically (using magic numbers in the superblock, or simply by trying different file system types; see **man mount** for details).

The following are some of the options you can use when mounting a file system with the command `mount` or by entering them in `/etc/fstab`:

- **remount.** This option causes file systems that are already mounted to be mounted again.

When you make a change to the options in `/etc/fstab`, you can use `remount` to incorporate the changes.
- **rw, ro.** These options indicate whether a file system should be writable (**rw**) or only readable (**ro**).
- **sync, async.** These options set synchronous (**sync**) or asynchronous (**async**) input and output in a file system. The default setting is `async`.
- **atime, noatime.** These options set whether the access time of a file is updated in the inode (**atime**) or not (**noatime**). The option `noatime` should improve the performance.
- **nodev, dev.** The **nodev** option prevents device files from being interpreted as such in the file system.
- **noexec, exec.** You can prohibit the execution of programs on a file system with the option **noexec**.
- **nosuid, suid.** The **nosuid** option ensures that the `suid` and `sgid` bits in the file system are ignored.

Some options only make sense in the file `/etc/fstab`. These options include the following:

- **auto, noauto.** File systems set with the option **noauto** in the file `/etc/fstab` are not mounted automatically when the system is booted.
- **user, nouser.** This option lets users mount the file system. Normally, this is a privilege of the user `root`.
- **defaults.** This option causes the default options `rw`, `suid`, `dev`, `exec`, `auto`, `nouser`, and `async` to be used.

The options **noauto** and **user** are usually combined for removable media such as floppy disk or CD-ROM drives.

Unmount a File System

Once a file system is mounted, you can use the **umount** command (without an “n”) to unmount the file system.

You can unmount the file system by using **umount** with the device or the mount point.

For example to unmount a CD file system mounted at `/media/cdrecorder`, you could enter one of the following:

- **umount /media/cdrecorder**

or

- **umount /dev/hdb**

In order to unmount the file system, no application or user may use the file system. If it is being used, Linux sees the file system as being “busy” and will refuse to unmount the file system.



To help determine the processes that are acting on a file or directory, you can use the **fuser** utility.

One way to make sure the file system is not busy is to enter **cd /** at the shell prompt before using the **umount** command. This command takes you to the root of the file system.

However, there might be times when the system (kernel) still sees the file system as busy, no matter what you try to do.

In these cases, you can enter **umount -f** to force the file system to unmount. However, we recommend using this only as a last resort, as there is probably a reason why the kernel thinks the file system is still mounted.

Exercise 4-1 *Configure Partitions on Your Hard Drive*

In this exercise, you practice creating partitions and file systems on them with YaST and **fdisk**. You also use command line tools to create file systems.

You will find this exercise in the workbook.

(End of Exercise)

Check a File System

Once you set up and begin using your Linux file system, you can monitor the status and health of the system by doing the following from the command line:

- Check lost+found (ext2 and ext3 only)
- Check and Repair File Systems (fsck)
- Check and Repair ext2/ext3 and ReiserFS (e2fsck and reiserfsck)
- Use Additional Tools to Manage File Systems

Check lost+found (ext2 and ext3 only)

The directory **lost+found** is a special feature of the ext2 and ext3 file system format. After a system crash, Linux automatically carries out a check of the complete file system. Files or file fragments to which a name can no longer be allocated are not simply deleted, but stored in this directory.

By reviewing the contents of this directory, you can try to reconstruct the original name and purpose of a file.

Check and Repair File Systems (fsck)

The command **fsck** lets you check and optionally repair one or more Linux file systems. Normally, fsck tries to run file systems on different physical disk drives in parallel to reduce the total amount of time to check all file systems.

If you do not specify a file system on the command line and do not specify the option **-A**, fsck defaults to checking file systems in `/etc/fstab` serially.

fsck is a frontend for the various file system checkers (**fsck.fstype**) available on the system. The fsck utility looks for the system-specific checker in /sbin/ first, then in /etc/fs/ and /etc/, and finally in the directories listed in the PATH environment variable.

To check a specific file system, use the following syntax:

fsck *device*

For example if you wanted to check the file system on /dev/hda2, you would enter **fsck /dev/hda2**.

Some options that are available with fsck include **-A** (walk through the /etc/fstab file and try to check all the file systems in one pass), **-N** (don't execute, just show what would be done), and **-V** (verbose output).

Check and Repair ext2/ext3 and ReiserFS (e2fsck and reiserfsck)

Switching off the Linux system without unmounting partitions (for example, when a power outage occurs) can lead to errors in the file system.

The next time you boot the system, the fact that the computer was not shut down correctly is detected and a file system check is performed. If errors are found in the file system, they are corrected, if possible. If not, the computer does not start up properly and you are prompted to enter the root password, together with a hint on how to correct the issue. In cases of severe file system damage, you may even have to resort to the rescue system to repair the system.

Depending on the file system type, you use either /sbin/e2fsck or /sbin/reiserfsck. These tools check the file system for a correct superblock (the block at the beginning of the partition containing information on the structure of the file system), faulty data blocks, or faulty allocation of data blocks.

A possible problem in the ext2 (or ext3) file system is damage to the superblock. You can first view the location of all copies of the superblock in the file system using **dumpe2fs**.

Then, with **e2fsck**, you can use one of the backup copies, as in the following:

```
e2fsck -f -b 32768 /dev/hda1
```

In this example, the superblock located at data block 32768 in the ext2 file system of the partition /dev/hda1 is used and the primary superblock is updated appropriately upon completion of the file system check.



With a block size of 4k, a backup copy of the superblock is stored every 32768 blocks.

With **reiserfsck**, the file system is subjected to a consistency check. The journal is checked to see if certain transactions need to be repeated. With the option **--fix-fixable**, errors such as wrong file sizes are fixed as soon as the file system is checked.

With an error in the binary tree, it is possible to have this rebuilt by entering **reiserfsck --rebuild-tree**.

Use Additional Tools to Manage File Systems

There are additional tools to administer various aspects of file systems.

tune2fs is used to adjust tunable filesystem parameters on ext2/ext3 file systems. Amongst these is the number of days or number of mounts a file system check is done. It is also used to add a label to the file system, or to add a journal to an ext2 file system, turning it into an ext3 file system.

reiserfstune is the corresponding tool for ReiserFS. See the reiserfstune manual page for options and uses for this tool.

resize2fs and **resize_reiserfs** are used to shrink or enlarge an ext2/3 and ReiserFS, respectively. **resize_reiserfs** can enlarge ReiserFS online. Shrinking file systems as well as enlarging ext2/3 can only be done while the file system is unmounted.



As stated before, when planning to manipulate partitions and file systems, back up your data first!

Exercise 4-2 Manage File Systems from the Command Line

In this exercise, you practice managing file systems from the command line.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 4 **Configure Logical Volume Manager (LVM) and Software RAID**

Logical volume manager (LVM) provides a higher-level view of the disk storage on a computer system than the traditional view of disks and partitions. This gives you much more flexibility in allocating storage space to applications and users.

After creating logical volumes with LVM, you can (within certain limits) resize and move logical volumes while they are still mounted and running.

You can also use LVM to manage logical volumes with names that make sense (such as “development” and “sales”) instead of physical disk names such as “sda” and “sdb.”

To configure a file system with LVM, you need to know the following:

- Use LVM Components
- Use LVM Features
- Configure Logical Volumes with YaST
- Configure LVM with Command Line Tools

The Linux Kernel is capable of combining hard disks to arrays with the RAID levels 0, 1, 5, and 6. Software RAID is covered in

- Manage Software RAID

Use LVM Components

Conventional partitioning of hard disks on a Linux file system is basically inflexible. When a partition is full, you usually have to move data to another medium before you can resize the partition, create a new file system, and copy the files back.

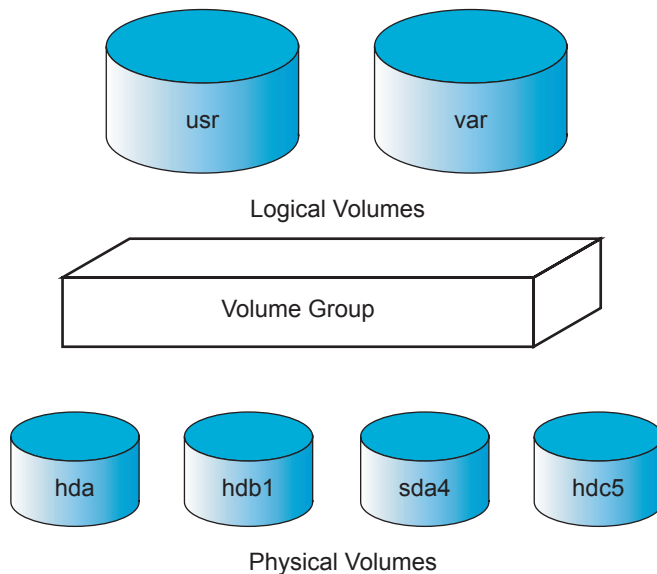
Normally, these changes cannot be implemented without changing adjacent partitions, whose contents also need to be backed up to other media and written to their original locations after the re-partitioning.

Because it is difficult to modify partitions on a running system, LVM was developed. It provides a virtual pool of memory space (called a *volume group*) from which logical volumes can be generated if needed. The operating system accesses these logical volumes like conventional physical partitions.

This approach lets you resize the physical media during operation without affecting the applications.

The basic structure of LVM includes the following components:

Figure 4-6



- **Physical volume.** A physical volume can be a partition or an entire hard disk.

- **Volume group.** A volume group consists of one or several physical volumes grouped together. The physical partitions can be spread over different hard disks. You can add hard disks or partitions to the volume group during operation whenever necessary.

The volume group can also be reduced in size by removing physical volumes (hard disks or partitions).

- **Logical volume.** A logical volume is a part of a volume group. A logical volume can be formatted and mounted like a physical partition.

You can think of volume groups as hard disks and logical volumes as partitions on those hard disks. The volume group can be split into several logical volumes that can be addressed with their device names (such as `/dev/system/usr`) like conventional partitions with theirs (`dev/hda1`).



Just as with other direct manipulations of the file system, a data backup should be made before configuring LVM.

Use LVM Features

LVM is useful for any computer, as it is very flexible when the need to adapt to changed needs for storage space arises.

The following are features of LVM that help you implement storage solutions:

- You can combine several hard disks or partitions into a large volume group.
- Provided there is unallocated space in the volume group, you can enlarge a logical volume when free space within the logical volume is exhausted. Resizing logical volumes is easier than resizing physical partitions.

- You can create extremely large logical volumes (Terabytes).
- You can add hard disks to the volume group in a running system, provided you have hot-swappable hardware capable of such actions.
- You can add logical volumes in a running system, provided there is free space in the volume group.
- You can use several hard disks with improved performance in the RAID 0 (striping) mode.
- There is no limit that is relevant in practice on the number of logical volumes (the limit in LVM version 1 was 256).
- The Snapshot feature enables consistent backups in the running system.

Configure Logical Volumes with YaST

The following are the basic steps for configuring logical volumes (LVM) with YaST:

- Define the LVM Partitions (Physical Volumes) on the Hard Drive
- Create the Volume Group and Logical Volumes
- Access the YaST Module `lvm_config`

Define the LVM Partitions (Physical Volumes) on the Hard Drive

During (or after) the installation of SUSE Linux Enterprise Server, you need to configure the LVM partition on the hard disk.

You can use YaST or `fdisk` to perform this task as described in “Configure Linux File System Partitions” on page 4-13.

For the File system ID, select **0x8E Linux LVM**. Do not create a file system on that partition.

Create the Volume Group and Logical Volumes

Select **LVM** in the YaST Expert Partitioner. The following appears:

Figure 4-7



You use this dialog to create a new logical volume group by entering the following:

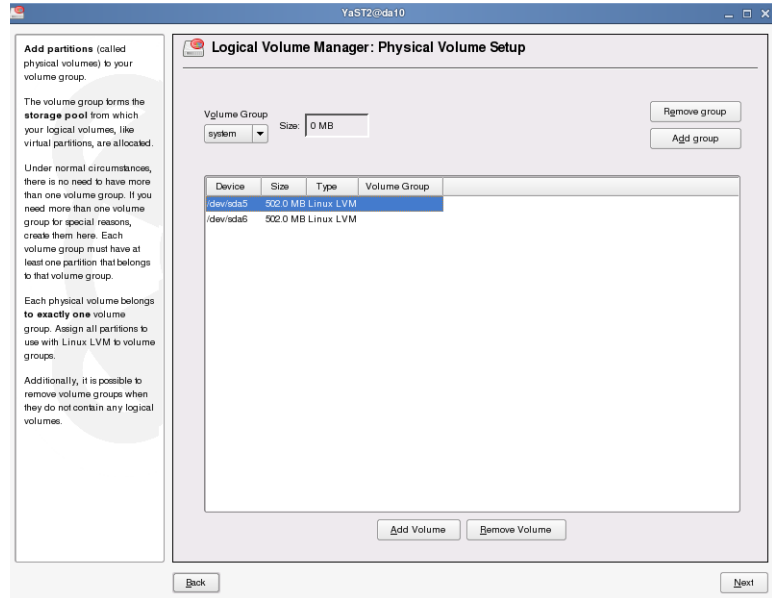
- **Volume Group Name.** Enter the name of your volume group.
- **Physical Extent Size.** The physical extent size defines the smallest unit of a logical volume group.

With LVM version 1, this also defined the maximum size of a logical volume. Entering a value 4 MB allowed logical volumes of 256 GB. With LVM2, this limitation does not exist anymore.

If you are not sure which values to enter, use the default settings.

After you select **OK**, the following appears:

Figure 4-8



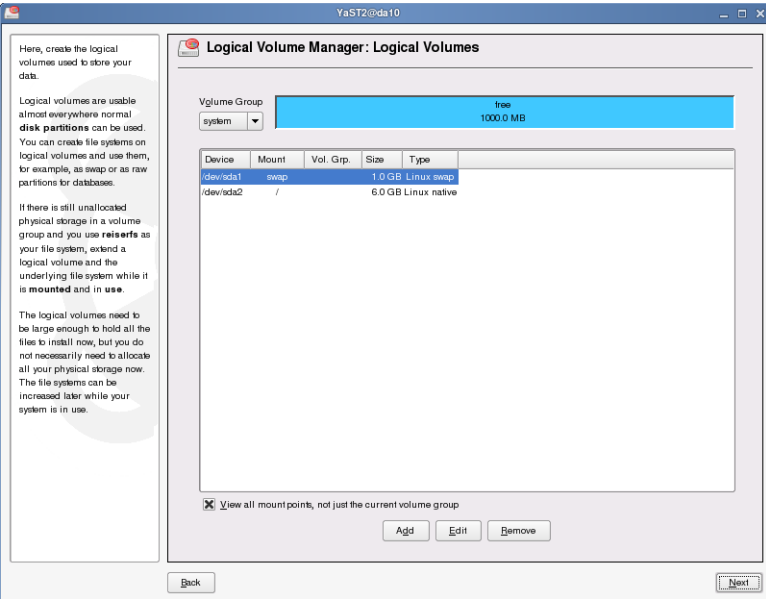
Here you can set up the physical volumes.

- **Volume Group.** Lets you select the volume group from the drop-down list that you want to add partitions to.
- **Size.** Displays the current size of the selected logical volume group.
- **Remove Group.** Deletes the currently selected volume group. You can delete empty groups only.
- **Add Group.** Adds a logical volume group.
- **Partition List.** Lets you select the partition you want to add to the volume group.
- **Add Volume.** Adds the selected partition to the volume group.

- **Remove Volume.** Removes the selected partition from the volume group.

Add physical volumes (these are usually partitions on a hard disk) to your volume group, and then select **Next** to continue. The following appears:

Figure 4-9

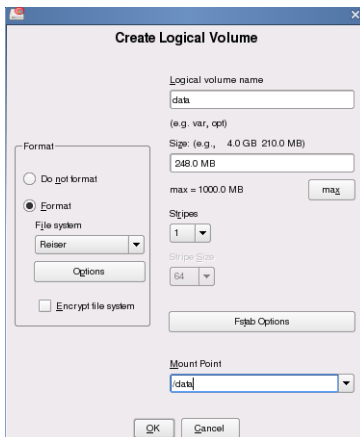


You can use the following to create logical volumes in your volume group:

- **Volume Group.** Allows you to select the volume group that you want to create partitions in.
- **Used/Available Space bar.** Displays the available space within the selected volume group.
- **Volume list.** Displays physical partitions and logical volumes in the system.

- **View all mount points, not just the current volume group.** When you select this option, all partitions and volumes that have entries in `/etc/fstab` are displayed. Otherwise, only the volumes in the selected volume group are displayed.
- **Add.** Adds a new logical volume to the volume group. When you select **Add**, the following appears:

Figure 4-10



This dialog lets you configure a logical volume using the same options available for creating a file system (see “Create a File System with YaST” on page 4-21).

In addition, you can enter a logical volume name, the maximum amount of space available (by selecting **max**), the number of stripes (equal or less than the number of disks), and the stripe size (if you configure more than one stripe).

Striping is only useful if you have two or more disks. It can increase performance by allowing parallel file system read and writes, but it also increases the risk of data loss. One failed disk can lead to data corruption in the whole volume group.

- **Edit.** Allows you to change the parameters of a selected volume.

The dialog to edit a volume has the same options as the dialog to create volumes (already described). You can also edit logical volumes directly from the partition list in the Expert Partitioner.

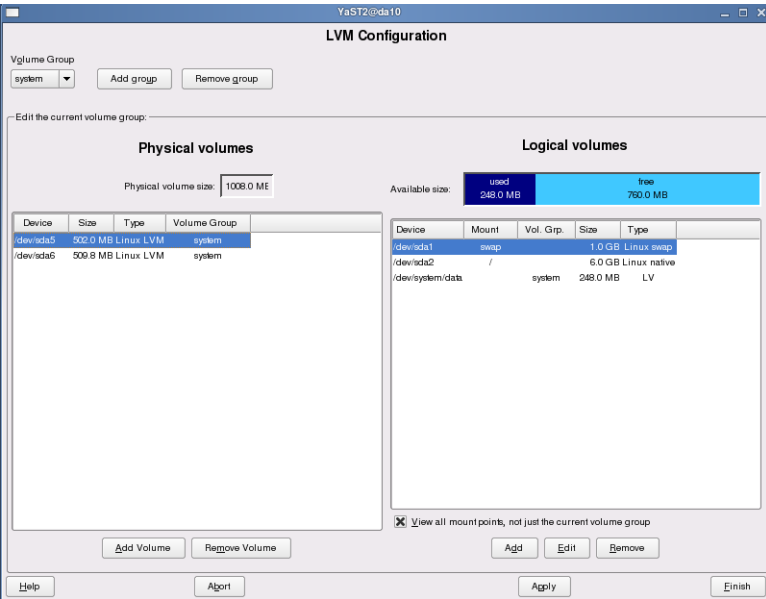
- **Remove.** Removes a selected volume. You can also remove logical volumes directly from the partition list in the Expert Partitioner.

When you are finished with the logical volume setup, select **Next** to save the settings and return to the Expert Partitioner.

Access the YaST Module lvm_config

To manage an existing LVM setup, you can access (as root) the YaST LVM configuration directly with **yast2 lvm_config**. It combines the configuration options for LVM in one dialog:

Figure 4-11



The configuration options are the same as those accessed by selecting LVM in the YaST Expert Partitioner.



For additional information on configuring LVM, see the LVM HOWTO at <http://tldp.org/HOWTO/LVM-HOWTO/>.

Configure LVM with Command Line Tools

Setting up LVM consists of several steps, with a dedicated tool for each:

- Tools to Administer Physical Volumes
- Tools to Administer Volume Groups
- Tools to Administer Logical Volumes

This is just a brief overview, not all available LVM tools are covered. To view the tools that come with LVM, enter **rpm -ql lvm2 | less** on a command line, and have a look at the corresponding manual pages for details on each of them.

Tools to Administer Physical Volumes

Partitions or entire disks can serve as physical volumes for LVM.

The ID of a partition used as part of LVM should be **Linux LVM**, **0x8e**. However the ID **0x83**, **Linux**, works as well.

To use an entire disk as physical volume, it may not contain a partition table. Overwrite any existing partition table with **dd**:

```
da10:~ # dd if=/dev/zero of=/dev/hdd bs=512 count=1
```

The next step is to initialize the partition for LVM. The tool to use is **pvcreate**:

```
da10:~ # pvcreate /dev/hda9
Physical volume "/dev/hda9" successfully created
```

pvscan shows the physical volumes and their use:

```
da10:~ # pvscan
PV /dev/hda9    lvm2 [242,95 MB]
Total: 1 [242,95 MB] / in use: 0 [0    ] / in no VG: 1 [242,95 MB]
```

The tool **pvmove** is used to move data from one physical volume to another (providing there is enough space), in order to remove a physical volume from LVM.

Tools to Administer Volume Groups

The tool **vgcreate** is used to create a new volume group. To create the volume group **system**, and add the physical volume **/dev/hda9** to it, enter:

```
da10:~ # vgcreate system /dev/hda9
Volume group "system" successfully created
da10:~ # pvscan
PV /dev/hda9    VG system    lvm2 [240,00 MB / 240,00 MB free]
Total: 1 [240,00 MB] / in use: 1 [240,00 MB] / in no VG: 0 [0    ]
```

pvscan shows the new situation.

To add further physical volumes to the group, use **vgexpand**. Removing unused physical volumes is done with **vgreduce** after shifting data from the physical volume scheduled for removal to other physical volumes using **pvmove**. **vgremove** removes a volume group, providing there are no logical volumes in the group.

Tools to Administer Logical Volumes

To create a logical volume, use **lvcreate**, specifying the size, the name for the logical volume, and the volume group:

```
da10:~ # lvcreate -L 100M -n data system
Logical volume "data" created
```

The next step is to create a file system within the logical volume and mount it:

```
da10:~ # lvscan
ACTIVE                '/dev/system/data' [100,00 MB] inherit
da10:~ # mkreiserfs /dev/system/data
mkreiserfs 3.6.19 (2003 www.namesys.com)
...
ReiserFS is successfully created on /dev/system/data.
da10:~ # mount /dev/system/data /data
```

As shown above, **lvscan** is used to view the logical volumes. It shows the device to use for the formatting and mounting.

lvextend is used to increase the size of a logical volume. After that you can increase the size of the file system on that logical volume to make use of the additional space.

Before you use **lvreduce** to reduce the size of a logical volume, you have to reduce the size of the file system. Only then reduce the size of the logical volume. If you cut off parts of the file system by simply reducing the size of the logical volume without shrinking the file system first, you will loose data.

Manage Software RAID

To manage software RAID (Redundant Array of Independent (or Inexpensive) Disks), select **RAID** in the YaST Expert Partitioner.

The purpose of RAID is to combine several hard disk partitions into one large virtual hard disk for optimizing performance and improving data security.

There are two types of RAID configurations:

- **Hardware RAID.** The hard disks are connected to a separate RAID controller. The operating system sees the combined hard disks as one device. No additional RAID configuration is necessary at the operating system level.
- **Software RAID.** Hard disks are combined by the operating system. The operating system sees every single disk and needs to be configured to use them as a RAID system.

In the past, hardware RAID provided better performance and data security than software RAID. However, with the current maturity of software RAID in the Linux kernel, software RAID provides comparable performance and data security.

In this section, you learn how to set up software RAID.

You combine hard disks according to RAID levels:

- **RAID 0.** This level improves the performance of your data access, however there is no redundancy in RAID 0. With RAID 0, two or more hard disks are pooled together (striping). Disk performance is very good, but the RAID system is vulnerable to a single point of failure. If one of the disks fails, all data is lost.
- **RAID 1.** This level provides enhanced security for your data because the data is copied to one or several hard disks. This is also known as *hard disk mirroring*. If one disk is destroyed, a copy of its contents is available on the other disk(s). Minimum number of disks (or partitions) required for RAID 1 is two.

- **RAID 5.** RAID 5 is an optimized compromise between RAID 0 and RAID 1 in terms of performance and redundancy. Data and a checksum are distributed across the hard disks. Minimum number of disks (or partitions) required for RAID 5 is three.

If one hard disk fails, it must be replaced as soon as possible to avoid the risk of losing data. The data on the failed disk is reconstructed on its replacement from the data on the remaining disks and the checksum. If more than one hard disk fails at the same time, the data on the disks is lost.

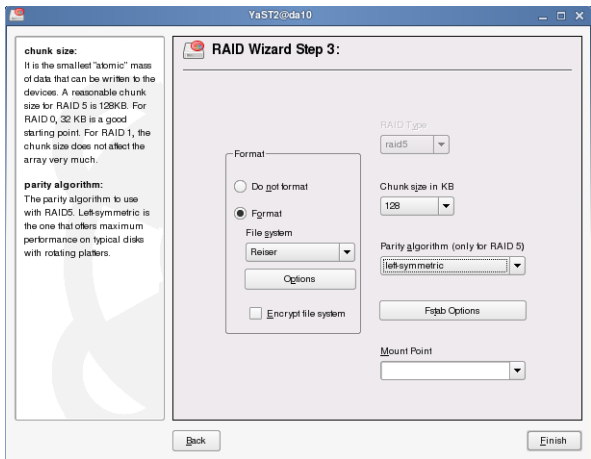
- **RAID 6.** RAID 6 is comparable to RAID 5, the difference being that 2 disks may fail without data loss. The minimum number of disks (or partitions) required for RAID 6 is four.

Using YaST you can set up RAID levels 0, 1, and 5 (RAID levels 2, 3, and 4 are not available with software RAID). To create software RAID with YaST, do the following:

- **Partition your hard disks.** For RAID 0 and RAID 1, at least 2 partitions on different disks are needed. RAID 5 requires at least 3 partitions. We recommend that you use only partitions of the same size.
- **Set up RAID.** Select **RAID** in the YaST Expert Partitioner to open a dialog to choose between the RAID levels 0, 1, and 5, and then add partitions to the new RAID.

Choose a file system and a mount point for your RAID. By changing the chunk size, which is explained in the help text in Figure 4-12, you can fine tune the RAID performance.

Figure 4-12



After finishing the configuration, the RAID partitions appear in the partition list of the Expert Partitioner.



For the purpose of testing, the partitions may reside on a single disk. However, this does not increase any performance or data security.



A RAID is no substitute for a data backup. A RAID does not, for instance, protect files from accidental deletion.

Exercise 4-3 Create Logical Volumes

In this exercise, you learn how to administer LVM using YaST.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Summary

Objective	Summary
1. Select a Linux File System	<p>Linux supports various file systems. Each file system has its particular strengths and weaknesses, which must be taken into account.</p> <p>File systems that keep a journal of transactions recover faster after a system crash or a power failure.</p>
2. Configure Linux File System Partitions	<p>A basic task of all system administrators is maintaining file system layouts. Under Linux, new partitions can be transparently grafted into existing file system structures using the mount command.</p> <p>In most cases, YaST proposes a reasonable partitioning scheme during installation. However, you can use YaST to customize partitioning during and after installation.</p> <p>To implement partitions on your SUSE Linux Enterprise Server, you learned about design guidelines for implementing partitions and how to administer partitions using YaST or command line tools.</p>

Objective	Summary
3. Manage Linux File Systems	<p>To perform basic Linux file system management tasks in SUSE Linux Enterprise Server, you learned how to use YaST and command line tools to create file systems on partitions.</p> <p>/etc/fstab is the configuration file that holds information about where each partition is to be mounted.</p> <p>mount is the command to attach file systems on partitions to the file system tree; umount detaches them.</p> <p>Various tools exist to monitor, repair, and tune file systems.</p>
4. Configure Logical Volume Manager (LVM) and Software RAID	<p>Logical volume management (LVM) provides a higher-level view of the disk storage on a computer system than the traditional view of disks and partitions.</p> <p>When you create logical volumes with LVM, you can resize and move logical volumes while partitions are still mounted and running.</p> <p>YaST can be used to create, edit or delete the components of LVM.</p> <p>Software RAID allows you to combine several disks to provide increased performance and redundancy.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 5 Manage System Initialization

In this section you learn how SUSE Linux Enterprise Server 10 boots and how to manage that process by setting kernel parameters, boot loader options, runlevels, and other system configurations.

SLES 10 differs from RHEL mainly in some of the start scripts, and the scripts are stored in different directories.

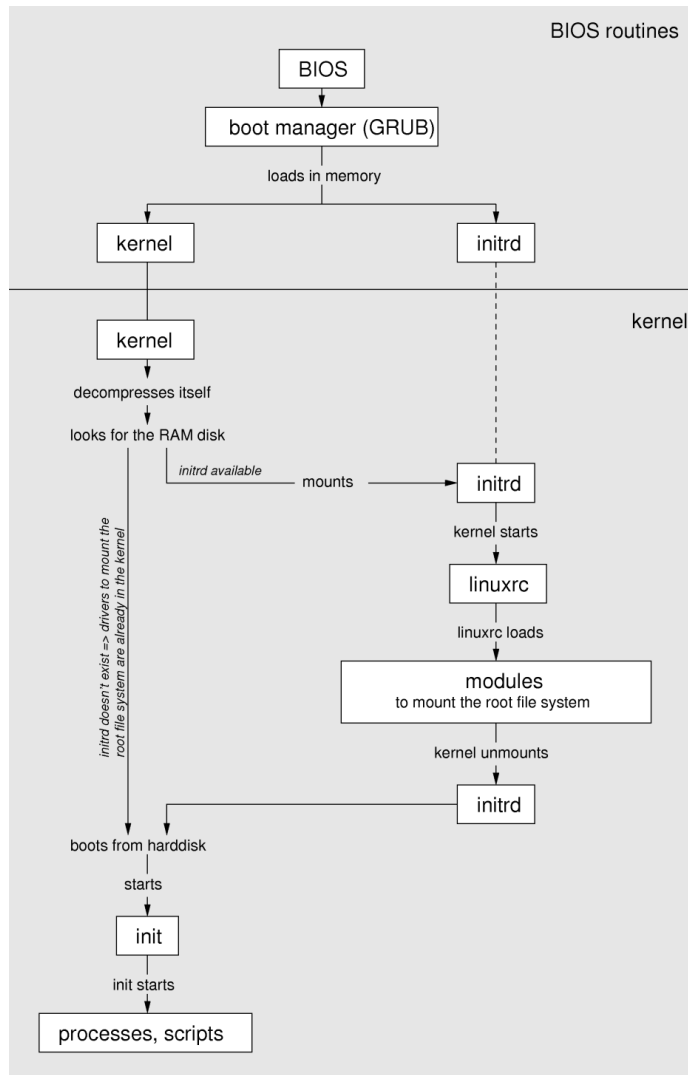
Objectives

1. Describe the Linux Load Procedure
2. GRUB (Grand Unified Bootloader)
3. Manage Runlevels

Objective 1 Describe the Linux Load Procedure

The following represents the basic steps of booting a computer with a Linux system installed:

Figure 5-1



1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

The following describe the process:

- BIOS and Boot Manager
- Kernel
- initramfs (Initial RAM File System)
- init

BIOS and Boot Manager

Tasks performed by the BIOS (Basic Input Output System) include performing a power-on self test, conducting the initial detection and setup of hardware, and accessing bootable devices (such as a CD or hard drive).

If the bootable device is a hard drive, BIOS also reads the MBR (master boot record). Using the code in the MBR, the BIOS starts the boot manager.

The *boot manager* (such as GRUB) loads the kernel and the `initrd` to memory and starts the kernel.

Kernel

The kernel (`/boot/vmlinuz`, which is a link to `/boot/vmlinuz-kernelversion`) uncompresses itself and then organizes and takes control of the continued booting of the system.

The kernel checks and sets the console (the BIOS registers of graphics cards and the screen output format), reads BIOS settings, and initializes basic hardware interfaces.

Next, the drivers, which are part of the kernel, probe existing hardware and initialize it accordingly.

The kernel controls the entire system, managing hardware access and allocating CPU time and memory to programs.

initramfs (Initial RAM File System)

initramfs is a cpio archive that the kernel can load to a RAM disk. It provides a minimal Linux environment that enables the execution of programs before the actual root file system is mounted. **initramfs** must always provide an executable named **init** that should execute the actual init program on the root file system for the boot process to proceed.

Former SUSE Linux versions used an initial RAM disk, **initrd**, instead. Despite the fact that the format changed, the file name is still **/boot/initrd**. **/boot/initrd** is a link to **/boot/initrd-*kernelversion***, the file that holds the gzipped cpio archive.

The kernel starts the program **init** contained in the **initramfs**. It is a shell script that, amongst other things, loads the kernel modules needed to mount the actual root file system, mounts the root file system and then finally starts **/sbin/init** from the root file system.

To look at the script **init** in **initramfs**, unpack the cpio archive:

```
da10:~ # mkdir /tmp/initramfs
da10:~ # cd /tmp/initramfs/
da10:/tmp/initramfs # gunzip -c /boot/initrd-2.6.16.14-6-smp | cpio -i
12765 blocks
da10:/tmp/initramfs # ls
bin bootsplash dev etc init lib proc root sbin sys tmp
da10:/tmp/initramfs # less init
```

The `initramfs` is created with the proper modules included, for instance those needed to access the file system, during installation. The modules to include are listed in the variable **INITRD_MODULES=** in `/etc/sysconfig/kernel`. If additional or different modules are needed, for instance due to a hardware change, you would edit the list of modules, and then rebuild the `initramfs`. The command is the same as the one to build an `initrd`, **mkinitrd**:

```
da10:~ # mkinitrd
Root device:      /dev/sda2 (mounted on / as reiserfs)
Module list:      piix aic7xxx sym53c8xx processor thermal
fan reiserfs edd (xennet xenblk)

Kernel image:     /boot/vmlinuz-2.6.16.14-6-smp
Initrd image:     /boot/initrd-2.6.16.14-6-smp
Shared libs:      lib/ld-2.4.so lib/libacl.so.1.1.0
lib/libattr.so.1.1.0 lib/libc-2.4.so lib/libdl-2.4.so
lib/libhistory.so.5.1 lib/libncurses.so.5.5
lib/libpthread-2.4.so lib/libreadline.so.5.1
lib/librt-2.4.so lib/libuuid.so.1.2

Driver modules:   ide-core ide-disk scsi_mod sd_mod piix
scsi_transport_spi aic7xxx sym53c8xx processor thermal fan
edd
Filesystem modules: reiserfs
Including:         initramfs fsck.reiserfs
Bootsplash:      SuSE-SLES (1024x768)
12765 blocks
```



The manual page for `mkinitrd` lists the parameters that can be passed to the `init` program in the `initramfs` via the kernel command line.

init

After checking the partitions and mounting the root file system, the program `init` located in `initramfs` starts `/sbin/init`, which boots the system with all its programs and configurations.

The init process is always assigned a process ID number of 1, and relies on the **/etc/inittab** file for configuration information on how to run the initialization process.

Once the init process starts, it begins by accessing the `/etc/init.d/boot` script. The `/etc/init.d/boot` script controls the start of services such as initializing disk quotas and mounting local file systems and executes the `Sxx`-scripts in `/etc/init.d/boot.d/`.

After the boot script has been completed, init starts the `/etc/init.d/rc` script which uses configured runlevels to start services and daemons.

Each runlevel has its own set of services that are initiated. For example, runlevel 5 includes the X Window components that run the Linux desktop.



For additional details on init, see “Manage Runlevels” on page 5-23.

Objective 2 GRUB (Grand Unified Bootloader)

RHEL and SLES 10 both use GRUB as their default boot loader.

To manage GRUB, the Grand Unified Bootloader, you need to know the following:

- What a Boot Manager Is
- Boot Managers in SUSE Linux
- Start the GRUB Shell
- Modify the GRUB Configuration File
- Configure GRUB with YaST
- Boot a System Directly into a Shell

What a Boot Manager Is

To boot a system, you need a program that can load the respective operating system into memory. This program, called the *boot loader*, loads the operating system kernel, which then loads the system.

After running the Power-On Self Test (POST), the PC BIOS searches various media configured in the BIOS for a boot loader. If it finds one, it turns control of the boot process over to the boot loader.

The boot loader then locates the operating system files on the hard drive and starts the operating system.

A *boot manager* is not only a boot loader, but it can handle several operating systems. If there is more than one operating system present, the boot manager presents a menu allowing you to select a specific operating system to be loaded.

Linux boot managers can be used to load Linux or other operating systems, such as Microsoft Windows.

GRUB is designed with the following 2-stage architecture:

- **Stage 1.** The first stage of a boot loader is usually installed in the master boot record (MBR) of the hard disk (first stage boot loader).

As the space in the MBR is limited to 446 bytes, this program code merely contains the information for loading the next stage.

Stage 1 can be installed in the MBR, in the boot sectors of partitions, or on a floppy disk.

- **Stage 2.** This stage usually contains the actual boot loader. The files of the second stage boot loader are located in the directory /boot/.

Boot Managers in SUSE Linux

SUSE Linux Enterprise Server provides 2 boot managers for the Linux environment: GRUB (GRand Unified Bootloader) and LILO (Linux LOader).

To gain a better understanding of these boot managers, you need to know the following:

- GRUB Boot Manager
- LILO Boot Manager
- Map Files, GRUB, and LILO

GRUB Boot Manager

GRUB is the standard boot manager in SUSE Linux Enterprise Server. The following are some special features of GRUB:

- **File system support.** Stage 2 includes file system drivers for ReiserFS, ext2, ext3, Minix, JFS, XFS, FAT, and FFS (BSD). For this reason, the boot manager can access files through filenames even before the operating system is loaded.

This feature is useful when the boot manager configuration is faulty and you have to search for and load the kernel.

- **Interactive control.** GRUB has its own shell that enables interactive control of the boot manager.

LILO Boot Manager

Because LILO is not the default boot manager of SUSE Linux Enterprise Server, it is only covered briefly in this objective.

The LILO configuration file is `/etc/lilo.conf`. Its structure is similar to that of the GRUB configuration file.



When you modify the configuration file `/etc/lilo.conf`, you need to enter the command **lilo** for the changes to be applied.

You also need to use the command `lilo` when moving the kernel or the `initrd` on your hard disk.

Map Files, GRUB, and LILO

The main obstacle for booting an operating system is that the kernel is usually a file within a file system on a partition on a disk. These concepts are unknown to the BIOS. To circumvent this, maps and map files were introduced.

These maps simply note the physical block numbers on the disk that comprise the logical files. When such a map is processed, the BIOS loads all the physical blocks in sequence as noted in the map, building the logical file in memory.

In contrast to LILO, which relies entirely on maps, GRUB tries to become independent from the fixed maps at an early stage. GRUB achieves this by means of the file system code, which enables access to files by using the path specification instead of the block numbers.



More information on GRUB and LILO can be found in the respective manual and info pages and in `/usr/share/doc/packages/grub` and `/usr/share/doc/packages/lilo`.

Start the GRUB Shell

As GRUB has its own shell, you can boot the system manually if the Linux system does not start due to an error in the boot manager.

There are two ways to start the GRUB shell:

- Start the GRUB Shell in the Running System
- Start the GRUB Shell at the Boot Prompt

Start the GRUB Shell in the Running System

To start the GRUB shell during operation, enter the command `grub` as root. The following appears:

```
GNU GRUB  version 0.94  (640K lower / 3072K upper memory)

[ Minimal BASH-like line editing is supported.  For the first word, TAB
  lists possible command completions.  Anywhere else TAB lists the
  possible completions of a device/filename. ]

grub>
```

As in a bash shell, you can complete GRUB shell commands with the Tab key. To find out which partition contains the kernel, enter the command **find**, as in the following:

```
grub> find /boot/vmlinuz
(hd0,2)

grub>
```

In this example, the kernel (`/boot/vmlinuz`) is located in the third partition of the first hard disk (`hd0,2`).

Close the GRUB shell by entering **quit**.

Start the GRUB Shell at the Boot Prompt

Start the GRUB shell at the boot prompt by doing the following:

1. From the graphical boot selection menu, press **Esc**.
A text-based menu appears.
2. Start the GRUB shell by typing **c** (US keyboard layout).

Modify the GRUB Configuration File

Configure GRUB by editing the file `/boot/grub/menu.lst`. The following is the general structure of the file:

- First, there are general options:
 - **color white/blue black/light-gray.** Colors of the boot manager menu.
 - **default 0.** The first entry (numbering from 0) is the default boot entry that starts automatically if no other entry is selected with the keyboard.
 - **timeout 8.** The default boot entry is started automatically after 8 seconds.
 - **gfxmenu (hd0,0)/boot/message.** This defines where the graphical menu is stored.
- The general options are followed by options for the various operating systems that can be booted with the GRUB.
 - **title *title*.** Each entry for an operating system begins with title.
 - **root (hd0,0).** The following entries are relative to this hard disk partition given in the syntax of GRUB, in this example the first partition on the first hard disk. With this entry it is not necessary to specify the partition on each of the following entries like kernel.

Note the following regarding the designations for hard disks and partitions:

GRUB does not distinguish between IDE and SCSI hard disks. The hard disk that is recognized by the BIOS as the first hard disk is designated as `hd0`, the second hard disk as `hd1`, and so on.

The first partition on the first hard disk is called `hd0,0`, the second partition `hd0,1`, and so on.

- ❑ **kernel /boot/vmlinuz.** This entry describes the kernel location, relative to the partition specified above. It is followed by kernel parameters, like `root=/dev/hda1`, `vga=normal`, etc.
- ❑ **initrd /boot/initrd.** This entry sets the location of the initial ramdisk (initramfs in SLES 10), relative to root (hd0,0) specified above. The initrd contains hardware drivers that are needed before the kernel can access the hard disk (such as a driver for the IDE or SCSI controller).

The following is an example of the configuration file
/boot/grub/menu.lst:

```
# Modified by YaST2. Last modification on Mon May 15 08:38:29 UTC 2006

color white/blue black/light-gray
default 0
timeout 8
gfxmenu (hd0,1)/boot/message

###Don't change this comment - YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 10
    root (hd0,1)
    kernel /boot/vmlinuz root=/dev/sda2 vga=0x317    resume=/dev/sda1
splash=silent showopts
    initrd /boot/initrd

###Don't change this comment - YaST2 identifier: Original name: floppy###
title Floppy
    chainloader (fd0)+1

###Don't change this comment - YaST2 identifier: Original name:
failsafe###
title Failsafe -- SUSE Linux Enterprise Server 10
    root (hd0,1)
    kernel /boot/vmlinuz root=/dev/sda2 vga=normal showopts ide=nodma
apm=off acpi=off noresume nosmp noapic maxcpus=0 edd=off 3
    initrd /boot/initrd
```

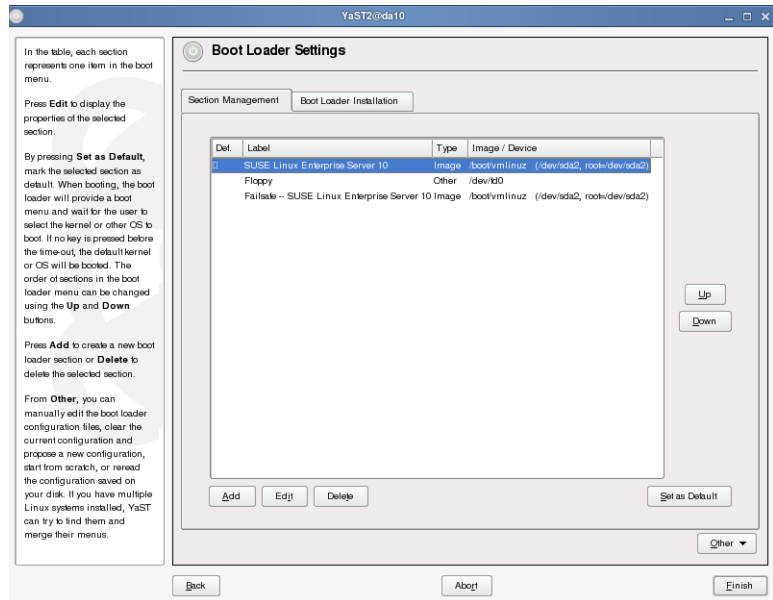
Configure GRUB with YaST

While you can use YaST (bootloader Configuration module) to simplify the configuration of the boot loader, you should not experiment with this module unless you understand the concepts behind it.

To start the YaST Boot Loader module, start YaST, enter the root password, then select **System > Boot Loader**, or start the Boot Loader module directly from a terminal window by entering as root **yast2 bootloader**.

The following appears:

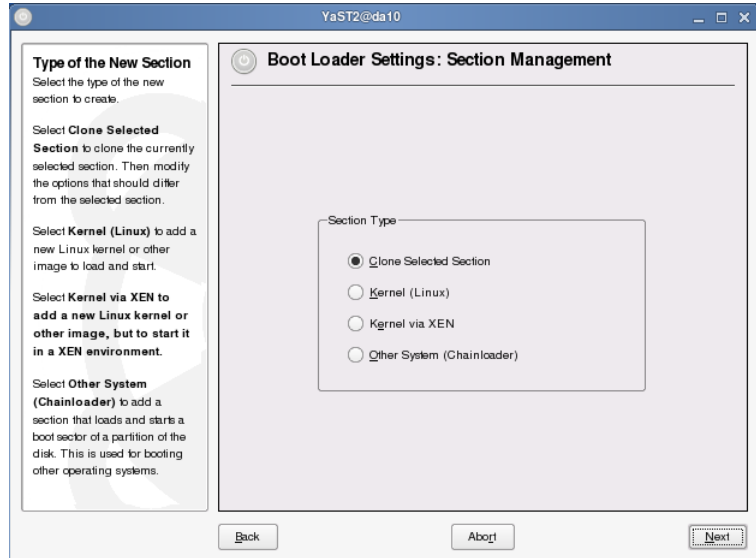
Figure 5-2



When the **Section Management** tab is selected, you see the current GRUB settings for your system. There is a **Def** (Default) column that indicates which entry is selected as the default when booting the system.

When you select Add, you are offered four choices:

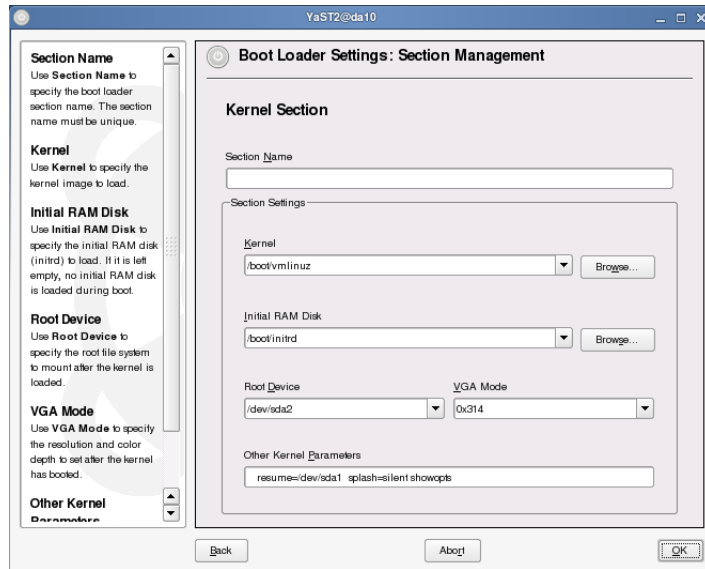
Figure 5-3



Each of the four **Section Types** is explained in the help text on the left.

When you select **Clone Selected Section** and click **Next**, the dialog is filled with the values from the selected section. With the two following options, the dialog is the same, but the lines are empty:

Figure 5-4



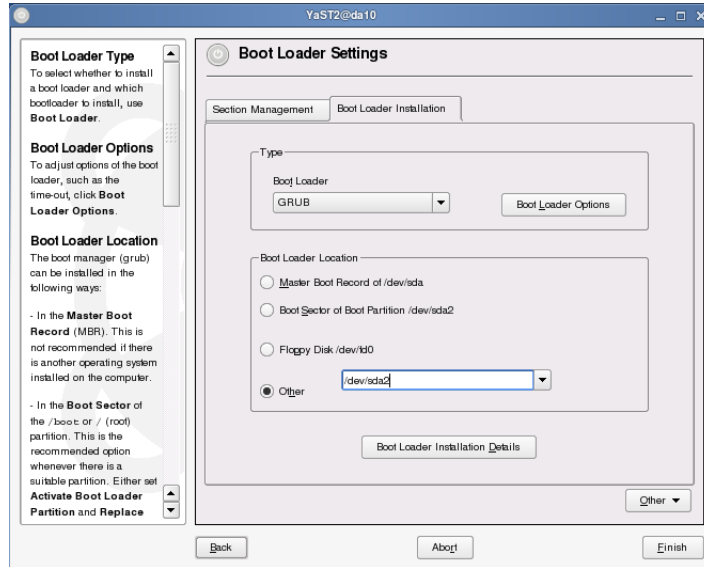
The dialog for the last choice, Other System (Chainloader), offers a line for a section name and a device from where to load another boot loader.

When you select **Edit** in the Boot Loader Settings dialog (Figure 5-2), the same dialogs opens up, where you can change the existing settings.

To delete an entry, select it and then click on **Delete**.

When you select the **Boot Loader Installation** tab, you see the following dialog:

Figure 5-5



- **Boot Loader Type.** You can use this option to switch between GRUB and LILO. A dialog lets you specify the way this change should be performed.
- **Boot Loader Location.** You can use this option to define whether to install the boot loader in the MBR, in the boot sector of the boot partition (if available), or on a floppy disk.
Use Others to specify a different location.
- **Boot Loader Installation Details.** This offers specialized configuration options, like activating a certain partition or changing the order of disks to correspond with the sequence in the BIOS.

- **Other.** When you select this, a drop-down menu with the following additional choices opens up:
 - **Edit Configuration Files.** Display and edit the configuration files (/boot/grub/device.map, /boot/grub/menu.lst, or /etc/grub.conf).

Note: /etc/grub.conf on SLES 10 contains the information needed to install the bootloader (**grub --batch </etc/grub.conf**), while the same file on RHEL has the same content as /boot/grub/menu.lst.
 - **Propose New Configuration.** This option generates a new configuration suggestion. Older Linux versions or other operating systems found on other partitions are included in the boot menu, enabling you to boot Linux or its old boot loader. The latter takes you to a second boot menu.
 - **Start from Scratch.** This option lets you create the entire configuration from scratch. No suggestions are generated.
 - **Reread Configuration from Disk.** If you already performed some changes and are not satisfied with the result, you can reload your current configuration with this option.
 - **Propose and Merge with Existing GRUB Menus.** If another operating system and an older Linux version are installed in other partitions, the menu is generated from an entry for the new SUSE Linux, an entry for the other system, and all entries of the old boot loader menu.

This procedure might take some time and is only available with GRUB.
 - **Restore MBR from Hard Disk.** The MBR saved on the hard disk is restored.
- 1. When you finish configuring the boot loader, save the configuration changes by selecting **Finish**.

Boot a System Directly into a Shell

The boot screen of the GRUB boot loader lets you pass parameters that modify the Linux kernel before the kernel is actually loaded.

At the bottom of the GRUB boot screen is a Boot Options field. To add a boot option, select an operating system and type the additional boot option in the Boot Options field.

One way to access a system that is not booting anymore is to set a different program for the init process. Normally, the Linux kernel tries to find a program with the name `init` and starts this program as the first process. All other processes are then started by `init`.

With the boot parameter `init=new_init_program`, you can change the first program loaded by the kernel. For example, by entering the boot parameter **`init=/bin/bash`**, the system is started directly into a bash shell. You are directly logged in as root without being asked for a password.



You could consider this somewhat equivalent to RHEL's runlevel 1, where you are not asked for the root password before getting a shell.

You can use this bash file to access the file system and to fix a misconfiguration.



The file systems are mounted as read-only after booting into a shell. To change configuration files, you need to remount the file system with the following command:

`mount -o remount,rw,sync -t filesystem_type device_name mount_point`

Entering **`exec /sbin/init`** at the bash prompt replaces the shell by the `init` program and continues the boot process until the default runlevel is reached.

If you want to prevent access to the machine as described above, you can change the boot configuration to require a password before the kernel command line can be edited.

In the file `/boot/grub/menu.lst`, the line

password *secret*

within the general options makes sure that the choices defined further below in the file (title SUSE SLES 10, etc.) can only be selected in unmodified form. The use of additional kernel parameters requires the password “*secret*”.

As the graphical boot menu could be used to circumvent the password feature, it is automatically disabled.

GRUB can also handle MD5-encrypted passwords that are generated as follows:

```
da10:~ # grub-md5-crypt
Password:
Retype password:
$1$FtTeK1$qaV.tOrzbg3EYAgVfNup40
```

This string can be copied to the file `/boot/grub/menu.lst`, with the following syntax:

password --md5 \$1\$FtTeK1\$qaV.tOrzbg3EYAgVfNup40

The parameter **lock** within a title section can be used to force the password query before these title entries can be selected.

```
title Floppy
    lock
    chainloader (fd0)+1
```

Selecting Floppy in the boot menu is now only possible after entering the password.

The parameter password can also be used in individual title entries to define a special password for those title entries.

Please note that the password feature only moderately enhances security, as it does not prevent booting the computer from another medium, like the SLES 10 rescue system, and accessing the files on the hard disk.



If you want to decide for each service (postfix, sshd, etc.) whether to start it or not during booting, use the parameter “confirm” at the boot prompt.

Exercise 5-1 Manage the Boot Loader

In this exercise, you practice booting into a shell and modifying /boot/grub/menu.lst.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 3 **Manage Runlevels**

Managing runlevels is an essential part of Linux system administration. In this objective, you learn what runlevels are, the role of the program `init`, and how to configure and change runlevels:

- The `init` Program and Linux Runlevels
- `init` Scripts and Runlevel Directories
- Change the Runlevel
- Compare Start Scripts between RHEL and SLES 10

The `init` Program and Linux Runlevels

- The `init` Program
- The Runlevels
- `init` Configuration File (`/etc/inittab`)

The `init` Program

The system is initialized by `/sbin/init`, which is started by the kernel as the first process of the system.

This process, or one of its child processes, starts all additional processes. In addition, because **`init`** is the last process running, it ensures that all other processes are correctly ended. This means that `init` controls the entire booting up and shutting down of the system.

Because of this position of priority, signal 9 (SIGKILL), with which all processes can normally be ended, has no effect on `init`.

The main configuration file of `init` is **`/etc/inittab`**. Various scripts are started by `init`, depending on entries in this file. All these scripts are located in the directory `/etc/init.d/`.

Part of the configuration in `/etc/inittab` is the runlevel the system uses after booting.

The Runlevels

In Linux, various runlevels define the state of the system. The following are the available runlevels:

Table 5-1

Command	Description
0	Halt
S	Used to boot into single-user mode (US keyboard layout)
1	Single-user mode
2	Multiuser mode without network server services
3	Multiuser mode with network
4	Not used
5	Multiuser mode with network and display manager
6	Reboot

The command **runlevel** displays the runlevel you are currently in (second number) and the previous runlevel (first number), as in the following:

```
da10:~ # runlevel
N 5
da10:~ #
```

init Configuration File (/etc/inittab)

To understand the contents of the file /etc/inittab, you need to know the following:

- inittab Syntax
- inittab Standard Entries

inittab Syntax

The following is the syntax of each line in the file /etc/inittab:

```
id:rl:action:process
```

The following describes the parameters:

- ***id.*** A unique name for the entry in /etc/inittab. It can be up to four characters long.
- ***rl.*** Refers to one or more runlevels in which this entry should be evaluated.
- ***action.*** Describes what init is to do.
- ***process.*** Is the process connected to this entry.

inittab Standard Entries

The first entry in the file /etc/inittab contains the following parameters:

```
id:5:initdefault:
```

The parameter initdefault signals to the init process which level it should bring the system to. The standard default runlevel is normally 3 or 5.

The next entry in `/etc/inittab` looks like this:

```
si::bootwait:/etc/init.d/boot
```

The parameter `bootwait` indicates to carry out this command while booting and wait until it has finished.

The next few entries describe the actions for runlevels 0 to 6:

```
10:0:wait:/etc/init.d/rc 0
11:1:wait:/etc/init.d/rc 1
12:2:wait:/etc/init.d/rc 2
13:3:wait:/etc/init.d/rc 3
#14:4:wait:/etc/init.d/rc 4
15:5:wait:/etc/init.d/rc 5
16:6:wait:/etc/init.d/rc 6

ls:S:wait:/etc/init.d/rc S
~~:S:respawn:/sbin/sulogin
```

The parameter `wait` means that when the system changes to the indicated level, the appropriate command is carried out and `init` waits until it has been completed. The parameter `also` means that further entries for the level are only performed after this process is completed.

The single user mode `S` is a special case, as it works even if the file `/etc/inittab` is missing. In such a case, enter `S` at the boot prompt when the computer starts. The command `sulogin` is started, which allows only the system administrator to log in. The parameter `respawn` indicates to `init` to wait for the end of the process and to then restart it.

`/etc/inittab` also defines the `Ctrl+Alt+Del` key combination for restarting:

```
ca::ctrlaltdel:/sbin/shutdown -r -t 4 now
```

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

The action `ctrlaltdel` is carried out by the `init` process only if these keys are pressed. If you do not want to allow this action, comment out (`#`) or remove the line.

The final large block of entries describes in which runlevels `getty` processes (login processes) are started:

```
1:2345:respawn:/sbin/mingetty --noclear tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

The `getty` processes provide the login prompt and in return expect a user name as input. They are started in runlevels 2, 3, and 5.



Runlevel 4 in the above example is ignored because the line that defines the actions for the runlevel is commented out earlier in the file:
(**#14:4:wait:/etc/init.d/rc 4**).

If a session ends, the processes are started again by `init`. If a line is disabled here, no further login is possible at the corresponding virtual console.



You should take great care when making changes to the file `/etc/inittab`. If the file is corrupted, the system will no longer boot correctly.

If an error does occur, first try entering `S` at the kernel command line in the GRUB boot menu. If this does not work, it is still possible to boot the system. Enter **`init=/bin/bash`** at the kernel command line in the GRUB boot menu.

In this way, the `init` process is replaced by a shell (so `inittab` is not read) and you can then repair the system manually.

When you changed `/etc/inittab`, use **`init q`** to have `init` reload its configuration.

1 **HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED**

init Scripts and Runlevel Directories

/etc/inittab defines the runlevel the system uses after booting is complete. The services that need to be started in a certain runlevel are not defined in /etc/inittab itself. These are configured by symbolic links in directories /etc/init.d/rcx.d/ which point to scripts in /etc/init.d/. To be able to manage runlevels, you need to understand the following:

- init Scripts
- Runlevel Symbolic Links
- How init Determines which Services to Start and Stop
- Activate and Deactivate Services for a Runlevel
- Activate and Deactivate Services for a Runlevel with YaST

init Scripts

The directory `/etc/init.d/` contains shell scripts that are used to perform certain tasks at boot up and start and stop services in the running system. The following shows some of the files in `/etc/init.d/`:

```
da10:~ # ls -al /etc/init.d/
total 635
drwxr-xr-x 11 root root 3336 May 24 13:40 .
drwxr-xr-x 77 root root 6712 May 25 13:19 ..
-rw-r--r-- 1 root root 1393 May 24 13:40 .depend.boot
-rw-r--r-- 1 root root 3465 May 24 13:40 .depend.start
-rw-r--r-- 1 root root 3002 May 24 13:40 .depend.stop
-rw-r--r-- 1 root root 482 Aug 25 2004 Makefile
-rw-r--r-- 1 root root 7827 May 10 18:17 README
-rwxr-xr-x 1 root root 1257 May 8 20:09 SuSEfirewall2_init
-rwxr-xr-x 1 root root 1650 May 8 20:09 SuSEfirewall2_setup
-rwxr-xr-x 1 root root 2696 May 8 20:29 aaeventd
-rwxr--r-- 1 root root 5729 May 8 20:15 acpid
-rwxr-xr-x 1 root root 5265 May 8 21:01 alsasound
-rwxr-xr-x 1 root root 3689 May 9 14:49 atd
-rwxr-xr-x 1 root root 6691 May 9 15:03 auditd
-rwxr--r-- 1 root root 9234 May 9 15:01 autofs
-rwxr-xr-x 1 root root 2967 Mar 14 13:40 autoyast
-rwxr-xr-x 1 root root 7072 Apr 20 15:02 boot
-rwxr-xr-x 1 root root 2792 May 8 20:29 boot.apparmor
...
```

The files `.depend.{boot,start,stop}` are created by `insserv` and contain dependencies that are used to determine the proper sequence for starting services when **RUN_PARALLEL** in `/etc/sysconfig/boot` is set to `yes`.

The shell scripts can be called up in the following ways:

- Directly by `init` when you boot the system, when the system is shut down, or when you stop the system with **Ctrl+Alt+Del**. Examples for these scripts are `/etc/init.d/boot` or `/etc/init.d/rc`.

- Indirectly by init when you change the runlevel. In this case, it is the script `/etc/init.d/rc` that calls the necessary scripts in the correct order and with the correct parameter during the runlevel change.
- Directly by *`/etc/init.d/script parameter`*.

You can also enter ***rcscript parameter*** if corresponding links are set in `/sbin/` or `/usr/sbin/`.

The following parameters may be used:

Table 5-2

Parameter	Description
start	Starts a service that is not running.
restart	Stops a running service and restarts it.
stop	Stops a running service.
reload	Rereads the configuration of the service without stopping and restarting the service itself.
force-reload	Reloads the configuration if the service supports this. Otherwise, it does the same thing as restart.
status	Displays the current status of the service.

When a script is called without parameters, a message informs you about the possible parameters.

Some of the more important scripts stored in `/etc/init.d/` are:

- **boot.** This script is started directly by init when the system starts. It is run once and once only. It evaluates the directory `/etc/init.d/boot.d/` and starts all the scripts linked by filenames with an “S” at the beginning of their names (see “Runlevel Symbolic Links” on page 5-31).

These scripts perform, for instance, the following tasks:

- Check the file systems

- ❑ Set up of LVM
- ❑ Delete unnecessary files in /var/lock/
- ❑ Set the system time
- ❑ Configure PnP hardware with the isapnp tools
- **boot.local.** This script includes additional commands to execute at boot before changing into a runlevel. You can add your own system extensions to this script.
- **halt.** This script is run if runlevel 0 or 6 is entered. It is called up either with the command halt (the system is completely shut down) or with the command reboot (the system is shut down and then rebooted).
- **rc.** This script is responsible for the correct change from one runlevel to another. It runs the stop scripts for the current runlevel, and then it runs the start scripts for the new one.
- **service.** Each service (like cron, apache2, cups) comes with a script allowing you to start and stop the service, to reload its configuration, or to view its status. To create your own scripts, you can use the file /etc/init.d/skeleton as a template.

Runlevel Symbolic Links

To enter a certain runlevel, init calls the script /etc/init.d/rc with the runlevel as parameter. This script examines the respective runlevel directory **/etc/init.d/rcx.d/** and starts and stops services depending on the links in this directory.

For each runlevel, there is a corresponding subdirectory in /etc/init.d/. For runlevel 1 it is /etc/init.d/rc1.d/, for runlevel 2 it is /etc/init.d/rc2.d/, and so on.

When you view the files in a directory such as `/etc/init.d/rc3.d/`, you see two kinds of files—those that start with a “K” and those that start with an “S”:

```
da10:~ # ls /etc/init.d/rc3.d/
K10cron          K17network       S07auditd
K10smbfs         K20haldaemon     S07portmap
K11nscd         K21acpid         S07splash_early
K11postfix       K21dbus          S08nfs
K12alsasound     K21fbset         S08nfsboot
K12boot.apparmor K21irq_balancer  S10alsasound
K12cups          K21random        S10boot.apparmor
K12microcode     K21resmgr        S10cups
K12powersaved   S01acpid         S10kbd
K12splash       S01dbus          S10microcode
K12sshd         S01fbset         S10powersaved
K14nfs          S01irq_balancer  S10splash
K14nfsboot      S01random        S10sshd
K15auditd       S01resmgr        S11nscd
K15portmap      S02haldaemon     S11postfix
K15splash_early S05network       S12cron
K16novell-zmd   S06novell-zmd    S12smbfs
K16slpd         S06slpd
K16syslog       S06syslog
```

The first letter is always followed by 2 digits and the name of a service. Whether a service is started in a specific runlevel depends on whether there are **Sxxservice** and **Kxxservice** files in the `/etc/init.d/rcx.d/` directory.

Entering **ls -l** in an `/etc/init.d/rcx.d/` directory indicates that these files are actually symbolic links pointing to service scripts in `/etc/init.d/` (as in the following):

```
da10:~ # ls -l /etc/init.d/rc3.d/
total 0
lrwxrwxrwx 1 root root 7 May 15 10:32 K10cron -> ../cron
lrwxrwxrwx 1 root root 8 May 15 10:48 K10smbfs -> ../smbfs
lrwxrwxrwx 1 root root 7 May 15 10:32 K11nscd -> ../nscd
lrwxrwxrwx 1 root root 10 May 15 10:32 K11postfix -> ../postfix
lrwxrwxrwx 1 root root 12 May 15 10:26 K12alsasound -> ../alsasound
...
lrwxrwxrwx 1 root root 7 May 15 10:31 S10sshd -> ../sshd
lrwxrwxrwx 1 root root 7 May 15 10:32 S11nscd -> ../nscd
lrwxrwxrwx 1 root root 10 May 15 10:32 S11postfix -> ../postfix
lrwxrwxrwx 1 root root 7 May 15 10:32 S12cron -> ../cron
lrwxrwxrwx 1 root root 8 May 15 10:48 S12smbfs -> ../smbfs
```

By using symbolic links in subdirectories only the version in `/etc/init.d/` needs to be modified in case of necessary changes to the script.

Usually, two links within a runlevel directory point to the same script. For example, if you enter

ls -l *network

in the `/etc/init.d/rc3.d/` directory, you see that two network links both point to the script `/etc/init.d/network`:

```
da10:~ # ls -l /etc/init.d/rc3.d/*network
lrwxrwxrwx 1 root root 10 May 15 10:23 /etc/init.d/rc3.d/K17network ->
../network
lrwxrwxrwx 1 root root 10 May 15 10:23 /etc/init.d/rc3.d/S05network ->
../network
```



Sometimes **Kxx** links are referred to as *kill scripts*, while **Sxx** links are referred to as *start scripts*. In fact, there are no separate scripts for starting and stopping services, but the script is either called with the parameter `stop` or with the parameter `start`.

How init Determines which Services to Start and Stop

You already know that a service is started with the parameter `start`, and stopped with the parameter `stop`. The same parameters are also used when changing from one runlevel to another.

When the runlevel is changed, `init` calls the script `rc` with the new runlevel as parameter, like **`/etc/init.d/rc 3`**. The script `/etc/init.d/rc` examines the directories `/etc/init.d/rccurrentrl.d/` and `/etc/init.d/rcnewrl.d/` and determines what to do.

Let's say we change from our current runlevel 5 to the new runlevel 3. There are three possibilities:

- There is a **Kxx** link for a certain service in `/etc/init.d/rc5.d/` and there is an **Sxx** link in `/etc/init.d/rc3.d/` for the same service.

In this case, the service is neither started nor stopped; the corresponding script in `/etc/init.d/` is not called at all.

- There is a **Kxx** link for a certain service in `/etc/init.d/rc5.d/` and there is no corresponding **Sxx** link in `/etc/init.d/rc3.d/`.

In this case, the script in `/etc/init.d/service` is called with the parameter `stop` and the service is stopped.

- There is an **Sxx** link in `/etc/init.d/rc3.d/` and there is no corresponding **Kxx** link for the service in `/etc/init.d/rc5.d/`.

In this case, the script in `/etc/init.d/service` is called with the parameter `start` and the service is started.

The number after the K or S determines the sequence in which the scripts are called.

Therefore script `K10cron` is called before script `K20haldaemon`, which means that `cron` is shut down before `haldaemon`.

Script `S05network` is called before `S11postfix`, which means that the service `network` starts before `postfix`. This is important if `postfix` depends on a running service `network`.

For example the following happens when you change from runlevel 3 to runlevel 5:

1. You tell init to change to a different runlevel by entering (as root) **init 5**.
2. init checks its configuration file (/etc/inittab) and determines it should start /etc/init.d/rc with the new runlevel (**5**) as a parameter.
3. rc calls the stop scripts (**Kxx**) of the current runlevel for those services for which there is no start script (**Sxx**) in the new runlevel.
4. The start scripts in the new runlevel for those services for which there was no kill script in the old runlevel are launched.

When changing to the same runlevel as the current runlevel, init only checks /etc/inittab for changes and starts the appropriate steps (such as starting a getty on another interface).

Activate and Deactivate Services for a Runlevel

Services are activated or deactivated in a runlevel by adding or removing the respective K**service and S**service links in the runlevel directories /etc/init.d/rcx.d/.

Although you could create symbolic links in the runlevel subdirectories yourself to modify services, an easier way is to edit the header of a script and then call **insserv**.

The INIT INFO block at the beginning of the script for a service describes in which runlevel the service should start or stop and what services should run as a prerequisite:

```
### BEGIN INIT INFO
# Provides:          syslog
# Required-Start:    network
# Should-Start:      earlysyslog
# Required-Stop:     network
# Default-Start:     2 3 5
# Default-Stop:
# Description:       Start the system logging daemons
### END INIT INFO
```

The INIT INFO block is used by the program `insserv` to determine in which runlevel subdirectories links need to be placed and what numbers need to be put after K and S.



For details on the program `insserv`, enter **`man 8 insserv`**.

The entry `Default-Start` determines in which runlevel directories links are to be placed. The entry `Required-Start` determines which services have to be started before the one being considered.

After editing the INIT INFO block, enter **`insserv -d service`** (default) to create the needed links and renumber the existing ones as needed.

To remove all links for a service (disabling the service), stop the service (if it is running) by entering **`/etc/init.d/service stop`**, and then enter **`insserv -r service`** (remove).

Within the INIT INFO block, the use of certain variables is possible. These are explained and defined in `/etc/insserv.conf`.

A tool with similar functionality is **chkconfig**. It can be used to disable or enable services and also to list which services are enabled in which runlevel. The following gives a brief overview on how to use **chkconfig**:

```
da10:~ # chkconfig cron
cron on
da10:~ # chkconfig cron -l
cron          0:off  1:off  2:on   3:on   4:off  5:on   6:off
da10:~ # chkconfig cron off
da10:~ # chkconfig cron -l
cron          0:off  1:off  2:off  3:off  4:off  5:off  6:off
da10:~ # chkconfig cron on
da10:~ # chkconfig -l
Makefile      0:off  1:off  2:off  3:off  4:off  5:off  6:off
SuSEfirewall2_init 0:off  1:off  2:off  3:off  4:off  5:off  6:off
SuSEfirewall2_setup 0:off  1:off  2:off  3:off  4:off  5:off  6:off
aaeventd      0:off  1:off  2:off  3:off  4:off  5:off  6:off
acpid         0:off  1:off  2:on   3:on   4:off  5:on   6:off
alsasound     0:off  1:off  2:on   3:on   4:off  5:on   6:off
atd           0:off  1:off  2:off  3:off  4:off  5:off  6:off
auditd        0:off  1:off  2:off  3:on   4:off  5:on   6:off
...
```

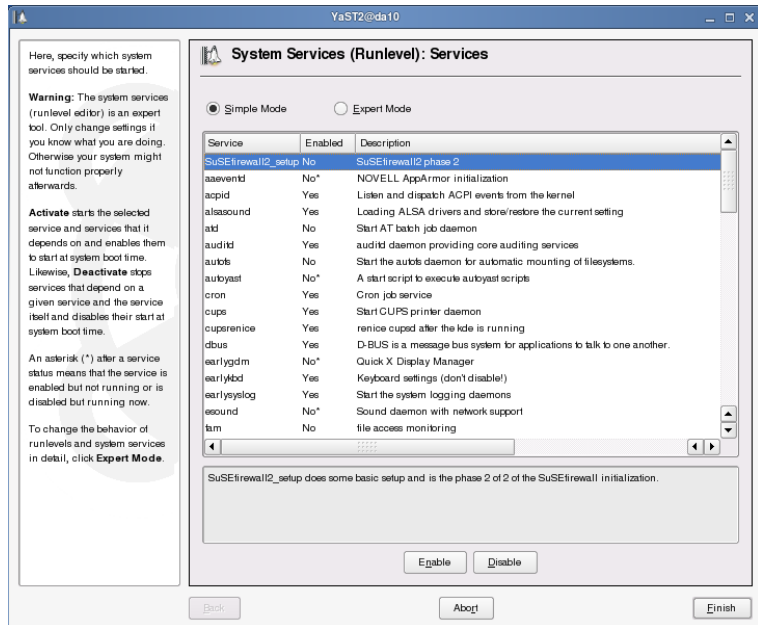
You can also use the YaST runlevel editor to set these links. We recommend that you either use **insserv/chkconfig** or **YaST**. Switching between methods can lead to errors.

Activate and Deactivate Services for a Runlevel with YaST

To configure runlevels with YaST, start the YaST Runlevel Editor module by starting **YaST** and then selecting **System > System Services (Runlevel)**, or open a terminal window and as root enter **yast2 runlevel**.

The following appears:

Figure 5-6



From this dialog, you can select from the following modes:

- **Simple Mode.** This mode displays a list of all available services and the current status of each service.

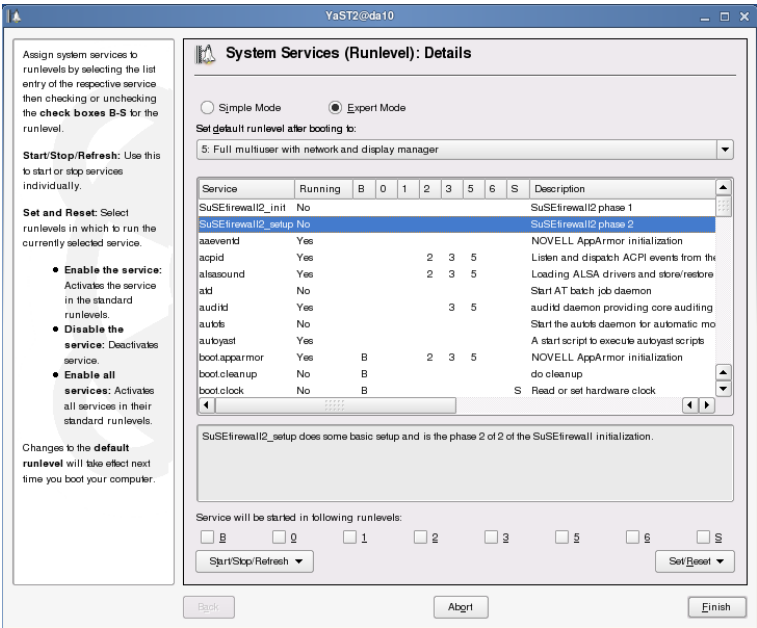
You can select a service, and then select **Enable** or **Disable**.

Selecting **Enable** starts the service (and services it depends on) and enables them to start at system boot time. Selecting **Disable** stops dependent services and the service itself and disables their start at system boot time.

- **Expert Mode.** This mode gives you control over the runlevels in which a service is started or stopped and lets you change the default runlevel.

Expert Mode looks like the following:

Figure 5-7



In this mode, the dialog displays the current default runlevel at the top. You can select a new default runlevel from the drop-down menu.

Normally, the default runlevel of a SUSE Linux system is runlevel 5 (full multiuser with network and graphical environment). A suitable alternative might be runlevel 3 (full multiuser with network). Runlevel 4 is initially undefined to allow creation of a custom runlevel.

Changes to the default runlevel take effect the next time you boot your computer.

To configure a service, select a service from the list, then from the options below the list, select the **runlevels** you want associated with the service.

The list includes the services and daemons available, indicates whether they are currently enabled on your system, and lists the runlevels currently assigned.

If you want a service activated after editing the runlevels, from the drop-down list select **Start now**, **Stop now**, or **Refresh status**.

You can use Refresh status to check the current status (if this has not been done automatically).

From the Set/Reset drop-down list, select one of the following:

- **Enable the service:** activates the service in the standard runlevels.
- **Disable the service:** deactivates the service.
- **Enable all services:** Activates all services in their standard runlevels.

When you finish configuring the runlevels, save the configuration by selecting **Finish**.

Remember that faulty runlevel settings can make a system unusable. Before applying your changes, make absolutely sure you know the impact of the changes.

Change the Runlevel

When starting the system, you can choose a runlevel different from the default runlevel defined in `/etc/inittab`. The runlevel can also be changed in the running system.

To change the runlevel, you need to understand:

- Change the Runlevel at Boot
- Manage Runlevels from the Command Line

Change the Runlevel at Boot

The standard runlevel is 3 or 5, as defined in the file `/etc/inittab` by the entry `initdefault`. However, it is also possible to boot to another runlevel by specifying the runlevel on the kernel command line of GRUB.

Any parameters that are not evaluated by the kernel itself are passed to `init` as parameters by the kernel. The desired runlevel is simply appended to the boot options already specified in GRUB (in the file `/boot/grub/menu.lst`), as in the following example:

```
root=/dev/hda1 vga=0x317 resume=/dev/hda2 splash=silent showopts 1
```

As root partition `/dev/hda1` is transmitted to the kernel, various parameters including the framebuffer are set, and the system boots to runlevel 1 (single user mode for administration).

Manage Runlevels from the Command Line

You can change to another runlevel once the system is running by using the command **init**. For example, you can change to runlevel 1 from a command line by entering **init 1**.

In the same way, you can change back to the standard runlevel where all programs needed for operation are run and where individual users can log in to the system.

For example, you can return to a full GUI desktop and network interface (runlevel 5) by entering **init 5**.



If the partition /usr of a system is mounted through NFS, you should not use runlevel 2 because NFS file systems are not available in this runlevel.

Like most modern operating systems, Linux reacts sensitively to being switched off without warning. If this happens, the file systems need to be checked and corrected before the system can be used again.

For this reason, the system should always be shut down properly. With the appropriate hardware, Linux can also switch off the machine as the last stage of shutting down.

You stop the system by entering **init 0**; you restart the system by entering **init 6**. The commands **halt** and **poweroff** are equivalent to **init 0**; the command **reboot** is equivalent to **init 6**.

The command **shutdown** shuts down the system after the specified time (+m: minutes from now; hh:mm: time in hours:minutes, when Linux should shut down; now: system is stopped immediately). The option -h causes a system halt, if you use the option -r instead the system is rebooted. Without options, it changes to runlevel 1 (single user mode).

The command **shutdown** controls the shutdown of the system in a special way, compared with the other stop commands. The command informs all users that the system will be shut down and does not allow other users to log in before it shuts down.

The command shutdown can also be supplied with a warning message, such as the following:

```
shutdown +5 The new hard drive has arrived
```

If a shutdown planned for a later time should not be carried out after all, you can revoke the shutdown by entering **shutdown -c**.

Compare Start Scripts between RHEL and SLES 10

The following table lists the files and directories with comparable functionality:

Table 5-3

RHEL 4	SLES 10
/etc/rc.d/init.d/	/etc/init.d/
/etc/rc.d/rcx.d/	/etc/init.d/rcx.d/
/etc/rc.d/init.d/service	/etc/init.d/service
/etc/rc.d/rc.sysinit	/etc/init.d/boot
/etc/rc.d/rc	/etc/init.d/rc

In SUSE Linux Enterprise Server 10, there is a symbolic link /etc/rc.d/ pointing to /etc/init.d/, as there is one on RHEL pointing from /etc/init.d/ to /etc/rc.d/init.d/. Therefore, administrators familiar with one system will easily find the files they are looking for in the other system.

Exercise 5-2 Manage Runlevels

In this exercise, you practice configuring runlevels.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Summary

Objective	Summary
1. Describe the Linux Load Procedure	<p>In this objective, you learned the following about the basic steps of booting a computer with a Linux system:</p> <ul style="list-style-type: none">■ BIOS and Boot Manager■ Kernel■ initramfs (Initial RAM File System)■ init
2. GRUB (Grand Unified Bootloader)	<p>The default boot manager in SLES 10 is GRUB. It is responsible for loading the operating system.</p> <p>Its configuration file is <code>/boot/grub/menu.lst</code>.</p> <p>The GRUB shell allows, amongst other things, to search for and view the content of files before the operating system is running.</p>

Objective	Summary
3. Manage Runlevels	<p>The initialization of the system is done by <code>/sbin/init</code>, which is started by the kernel as the first process of the system.</p> <p>The central configuration file of <code>init</code> is <code>/etc/inittab</code>.</p> <p>Various scripts are started by <code>init</code>. These scripts are located in the directory <code>/etc/init.d/</code>.</p> <p>In Linux, various runlevels define the state of the system.</p> <p>The system administrator can change to another runlevel with the command <code>init</code>.</p> <p>The command <code>runlevel</code> displays the previous and the current runlevel.</p>

SECTION 6 Configure Mail and Web Services

This section covers two of the more frequently used services.

As Postfix is the default mail server on SLES 10 (RHEL 4 uses Sendmail as default), Postfix is covered in more detail.

Objectives

1. Postfix
2. Apache Web Server

Objective 1 **Postfix**

Both, RHEL4 and SUSE Linux Enterprise Server 10, allow to choose between Sendmail and Postfix as mail server. While Sendmail is the default mail server under Redhat, the default mail server in SUSE Linux Enterprise Server 10 is Postfix.

Postfix was written by Wietse Venema as an alternative to the well-known Mail Transfer Agent (MTA) Sendmail with the following goals:

- It should be a fast mailer.
- It should be easy to administer.
- It should be secure.
- It should be compatible with Sendmail.

This objective covers the following topics:

- Understand the Architecture and Components of Postfix
- Configure Postfix
- Use Postfix Tools

Understand the Architecture and Components of Postfix

Wietse Venema met his Postfix design goals using a series of modular function units.

Unlike Sendmail, Postfix is not a large monolithic program block. Instead, it consists of a variety of small programs, each of which is allocated a specific task (for example, accepting an email).

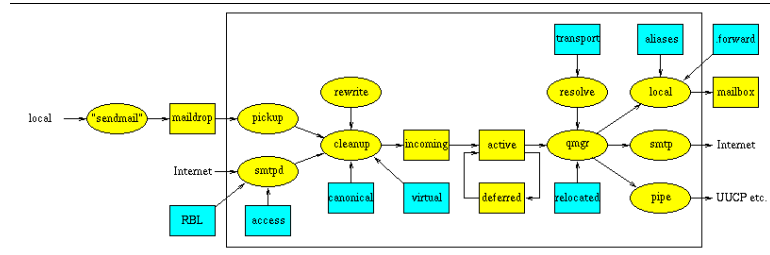
This modularization makes the system more transparent.

The individual components are easier to administer, facilitating further development of Postfix.

The following figure, taken from the original Postfix documentation, shows a rough summary of the modularization of Postfix.

Modules that are not covered at this stage are in </usr/share/doc/packages/postfix/html/OVERVIEW.html>.

Figure 6-1



Individual Postfix processes are represented in the diagram by ellipses. Dark squares stand for lookup tables and light squares represent mail queues or mailboxes.

For security reasons, Postfix works with four mail queues. For every mail queue, there is a directory bearing the same name under `/var/spool/postfix/`.

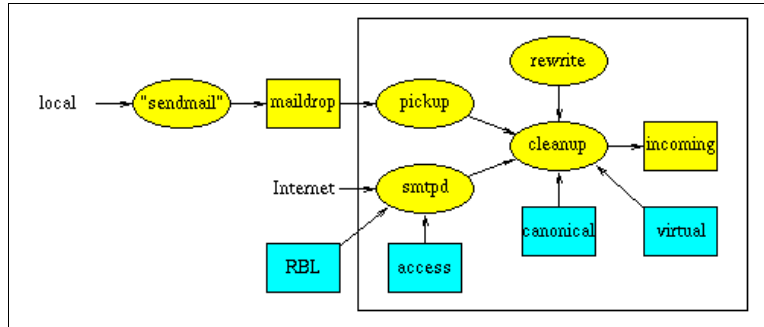
The functions of the queues and the Postfix files are described in

- Process of Inbound Email
- Process of Outbound Email
- Components of the Postfix Program Package

Process of Inbound Email

The following figure shows how an email can reach Postfix and how it is processed.

Figure 6-2



The following describe these processes:

- Email Received Locally
- Email Received over the Network

Email Received Locally

Postfix uses the **postdrop** command to place an email sent locally into the maildrop queue before it is picked up by the pickup daemon.

The pickup daemon checks it for content, size, and other factors based on rules; then it passes the email to the cleanup daemon.

The cleanup daemon does the following:

- Inserts missing header lines (Resent:, From:, To:, Message-ID:, Date:) in the email (if the mail was written with telnet)
- Deletes double recipient addresses

- Uses the trivial-rewrite daemon (/usr/lib/postfix/trivial-rewrite) to convert the email address in the header to the *user@fully-qualified-domain* convention
- Writes data in the header according to the rules in the lookup tables /etc/postfix/canonical and /etc/postfix/virtual

After this, the email is copied to the incoming queue and the queue manager /usr/lib/postfix/qmgr is informed of the arrival of this email.

Email Received over the Network

Email received over the Internet or LAN is accepted by the daemon, smtpd. smtpd checks the email for content, size, and other factors before passing it to the cleanup daemon.

The cleanup daemon does the following:

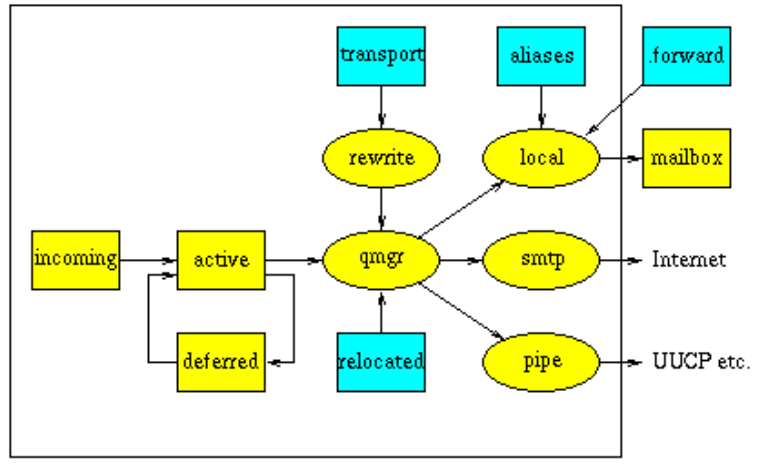
- Replaces missing header lines (Resent:, From:, To:, Message-ID:, Date:) in the email
- Deletes double recipient addresses
- Uses the trivial-rewrite (/usr/lib/postfix/trivial-rewrite) daemon to convert the email address in the header to the *user@fully-qualified-domain* convention
- Writes data in the header according to the rules of the lookup tables /etc/postfix/canonical and /etc/postfix/virtual

Then the email is copied to the incoming queue and the queue manager /usr/lib/postfix/qmgr is informed of the arrival of this email.

Process of Outbound Email

The following figure shows how an email is handled by Postfix before it leaves the system to be delivered to its destination:

Figure 6-3



The following topics describe this process:

- Deliver Email to Local Users
- Deliver Email to Users on Remote Systems
- Process Undeliverable Emails

Deliver Email to Local Users

The queue manager fetches an email from the incoming queue and copies it to the active queue as soon as the active queue contains no other emails.

The trivial-rewrite daemon takes over the checking procedure based on the lookup table `/etc/postfix/transport` to see whether the recipient of the email is on the local system or a remote system.

If this daemon decides the email should be delivered locally, the queue manager orders the local delivery service (`/usr/lib/postfix/local`) to deliver the email to the recipient's mailbox, taking into account the alias database (`/etc/aliases`) as well as any forward files of the user (`~/.forward`).

The local daemon can also be configured to have mail delivered by external programs, such as Procmail.

Deliver Email to Users on Remote Systems

The queue manager fetches an email from the incoming queue and copies it to the active queue, as soon as the active queue is empty.

The trivial-rewrite daemon uses the `/etc/postfix/transport` lookup table to see if the recipient of the email is on the local system or on a remote system.

If the daemon decides the email should be delivered to a remote system, the queue manager activates the SMTP service to deliver the email.

The SMTP service tries to find the mail exchanger specified for the target host; then it delivers the email to the mail exchanger for the recipient host.

Process Undeliverable Emails

Emails that cannot be delivered are removed from the active queue by the queue manager and copied to the deferred queue.

The queue manager then copies this email at regular intervals from the deferred queue back to the active queue and tries again to deliver the email.

Components of the Postfix Program Package

During the Postfix installation, files are saved to various locations on a SUSE Linux Enterprise Server 10 system. These locations can be grouped according to the following criteria:

- **/etc/aliases**. This is the only file in **/etc/**. It has the same format as the aliases file for the MTA Sendmail and contains local address aliases.
- **/etc/postfix/**. All the configuration files defining Postfix mail processing are in this directory.

Normally, the Postfix administrator is the only one who can make changes to these files.

- **/usr/lib/postfix/**. This directory contains all the programs needed directly by Postfix. To be more precise, these are the Postfix binaries.

These programs are not accessed directly by the system administrator.

- **/usr/sbin/**. This directory contains the administration programs for maintaining and manually controlling Postfix.

An administrator uses these programs during maintenance work.

- **/usr/bin/**. This directory contains symbolic links with the names **mailq** and **newaliases**.

Both links point to the program **/usr/sbin/sendmail** that provides a Sendmail-compatible administration interface for Postfix.

- **/var/spool/postfix/**. This directory contains the queue directories for Postfix and the directories **etc/** and **lib/** for Postfix processes that run in a chroot environment.

If the variables **POSTFIX_CHROOT** and **POSTFIX_UPDATE_CHROOT_JAIL** in **/etc/sysconfig/postfix** are set to **yes**, these two directories are set up by

SuSEconfig --module postfix

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

- **/usr/share/man/man[1|5|8]/**. These directories contain the manual pages for the Postfix binaries, for the configuration files, and the administration programs.
- **/usr/share/doc/packages/postfix/**. This contains documentation for Postfix.

The subdirectory `html/` contains a detailed HTML description of Postfix and a very useful FAQ.

Configure Postfix

This objective covers the following topics:

- Configure the Postfix Master Daemon
- Configure Global Settings
- Configure General Scenarios
- Configure the Lookup Tables

Configure the Postfix Master Daemon

The Postfix master daemon `/usr/lib/postfix/master` is started directly by Postfix when the system is booted and is terminated only when the system goes down or if Postfix ends.

The Postfix master daemon is normally configured once only when as the email system is set up, and is usually never changed.

The master daemon, which monitors the entire mail system,

- Controls and monitors individual Postfix processes.
- Adheres to configured resource limits, which were defined in the file `master.cf`.
- Restarts killed Postfix processes.

The Postfix master daemon is configured in the file `/etc/postfix/master.cf`. Each line in the file contains an entry for one Postfix process.

The behavior of each process is defined by the configuration in the respective line:

```
#
=====
# service type  private unpriv  chroot  wakeup  maxproc command + args
#               (yes)     (yes)   (yes)   (never) (100)
#
=====
smtp      inet  n       -       n       -       -       smtpd
#smtps    inet  n       -       n       -       -       smtpd
#  -o smtpd_tls_wrappermode=yes -o smtpd_sasl_auth_enable=yes
#submission inet  n       -       n       -       -       smtpd
#  -o smtpd_enforce_tls=yes -o smtpd_sasl_auth_enable=yes -o
smtpd_etrn_restrict
ions=reject
#628      inet  n       -       n       -       -       qmqpd
pickup    fifo  n       -       n       60      1       pickup
cleanup   unix  n       -       n       -       0       cleanup
qmgr       fifo  n       -       n       300     1       qmgr
#qmgr      fifo  n       -       n       300     1       oqmgr
#tlsmgr    fifo  -       -       n       300     1       tlsmgr
rewrite    unix  -       -       n       -       -       trivial-rewrite
bounce     unix  -       -       n       -       0       bounce
defer      unix  -       -       n       -       0       bounce
trace      unix  -       -       n       -       0       bounce
verify     unix  -       -       n       -       1       verify
flush      unix  n       -       n       1000?   0       flush
proxymap   unix  -       -       n       -       -       proxymap
smtp       unix  -       -       n       -       -       smtp
relay      unix  -       -       n       -       -       smtp
#  -o smtp_helo_timeout=5 -o smtp_connect_timeout=5
showq      unix  n       -       n       -       -       showq
error      unix  -       -       n       -       -       error
local      unix  -       n       n       -       -       local
virtual    unix  -       n       n       -       -       virtual
lmtpl      unix  -       -       n       -       -       lmtpl
anvil      unix  -       -       n       -       1       anvil
#localhost:10025 inet  n       -       n       -       -       -       smtpd -o
content
_filter=
...
```

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

```
...
#
# Interfaces to non-Postfix software. Be sure to examine the manual
# pages of the non-Postfix software to find out what options it wants.
#
# maildrop. See the Postfix MAILDROP_README file for details.
#
maildrop  unix  -      n      n      -      -      pipe
         flags=DRhu user=vmail argv=/usr/local/bin/maildrop -d ${recipient}
cyrus     unix  -      n      n      -      -      pipe
         user=cyrus argv=/usr/lib/cyrus/bin/deliver -e -r ${sender} -m
${extension} ${u
ser}
uucp      unix  -      n      n      -      -      pipe
         flags=Fqhu user=uucp argv=uux -r -n -z -a$sender - $nexthop!rmail
($recipient)
ifmail    unix  -      n      n      -      -      pipe
         flags=F user=ftn argv=/usr/lib/ifmail/ifmail -r $nexthop ($recipient)
bsmtp     unix  -      n      n      -      -      pipe
         flags=Fq. user=foo argv=/usr/local/sbin/bsmtp -f $sender $nexthop
$recipient
vscan     unix  -      n      n      -      10      pipe
         user=vscan argv=/usr/sbin/amavis ${sender} ${recipient}
procmail  unix  -      n      n      -      -      pipe
         flags=R user=nobody argv=/usr/bin/procmail -t -m /etc/procmailrc
${sender} ${r
ecipient}
```

If an entry in the file is too long for a specific service, this entry can be continued in the following lines by adding an empty space at the beginning of the following line; for example:

```
procmail  unix  -      n      n      -      -      pipe
         flags=R user=nobody argv=/usr/bin/procmail -t -m /etc/procmailrc
${sender} ${recipient}
```

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

The meaning of individual fields in a configuration line and their possible values are listed below.

Default values, if any, are listed in the description. If an entry is set to “—”, the default value is used.

- **service.** The name of the Postfix process.

An entry for a service that is controlled by the inet daemon can be specified in the form **host:port**.

inet is the service that controls who can connect to your computer and which services they can use.

An entry for the SMTP service could be

localhost:smtp

This entry would start the Postfix process /usr/lib/postfix/smtpd in such a way that it only receives email messages on port 25 of the loopback interface (if this port is entered correctly in the file /etc/services).

The host prefix and the following colon are optional.

- **type.** Allows you to specify a connection type.

Possible entries are

- **inet** for Internet sockets (TCP/UDP)
- **unix** for UNIX domain sockets (only for local communication)
- **fifo** (first in, first out) for named pipes

- **private.** Configures access to the service.

The value **y** (yes) only defines access to this service from the mail system.

The entry **n** (no) also allows access to this service for components outside the mail system. For services of the type inet, the value **n** must always be set.

The default value is **y**.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

- **unpriv.** Configures the UID under which this service is running.

With the value **y** (yes), the configured service runs under the unprivileged user configured in the file `/etc/postfix/main.cf` with the variable `mail_owner` (as a rule, the user `postfix`).

If this value is set to **n** (no), the service runs with root privileges - with the UID 0.

The default value is **y**.

- **chroot.** Specifies the chroot behavior of the service.

The value **y** (yes) causes the service to be started in a chroot environment.

The root path of this environment is defined in the variable `queue_directory` in the file `/etc/postfix/main.cf` (this is normally the directory `/var/spool/postfix/`).

The default value is **y**.

- **wakeup.** Runs the service again after the given number of seconds have expired.

The default value of **0** deactivates this function for the service.

Currently only the pickup daemon and the queue manager use this function.

The default value is **0** (never).

- **maxproc.** Defines the maximum number of processes that can be run simultaneously.

The default value is defined in the variable `default_process_limit` in the file `/etc/postfix/main.cf`.

The default value is **100**.

- **command + args.** Configures the command to run, including the required arguments.

The path name of the command to run is relative to the directory defined in the file `/etc/postfix/main.cf` via the variable `daemon_directory` (this is normally the directory `/usr/lib/postfix/`).

If one or more **-v** arguments are given, the debugging level is increased for the given command.

Specifying the **-D** argument allows debugging by using the debugging command, specified in the file `/etc/postfix/main.cf` by the variable `debugger_command`.

Configure Global Settings

All other configuration definitions (apart from the configuration of processing rules in lookup tables) are set in the following file:

`/etc/postfix/main.cf`

On SUSE Linux Enterprise Server 10, the most common parameters of this file can be modified using variables in the files

- **`/etc/sysconfig/mail`**
and
- **`/etc/sysconfig/postfix`**

Postfix is one of the last services that needs **SuSEconfig** to run for generation of the actual configuration files from files located in `/etc/sysconfig/`.

The file `/etc/sysconfig/mail` is used for general configurations that are not specific for Postfix and also used for Sendmail: For the MTA to operate correctly, you have to do the following in the file `/etc/sysconfig/mail`:

1. The fully qualified domain name (FQDN) must be entered in the variable `FROM_HEADER`.

If this variable is not set, the host name (FQDN) will be used.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

2. The variable `SMTPD_LISTEN_REMOTE` should be set to **yes** and Postfix will listen on port 25 for arriving mails.

Otherwise, only email from the local host will be accepted.

By means of the `/sbin/SuSEconfig` script, both settings and the entries in the file `/etc/sysconfig/postfix` are translated into suitable parameters in the file `/etc/postfix/main.cf`.

If you do not want **SuSEconfig** to generate this configuration file, set the variable `MAIL_CREATE_CONFIG` in the file `/etc/sysconfig/mail` to **no**.

To configure Postfix, you need to know how to do the following:

- Configure Postfix with `/etc/sysconfig/postfix`
- Configure Postfix with `/etc/postfix/main.cf`

Configure Postfix with `/etc/sysconfig/postfix`

Modifications in the file `/etc/sysconfig/postfix` are only adopted in the file `/etc/postfix/main.cf` and, in some cases, in the file `/etc/postfix/master.cf` after the execution of `/sbin/SuSEconfig` or the `SuSEconfig` module for Postfix:

- `/sbin/conf.d/SuSEconfig.postfix`
- or
- `/sbin/SuSEconfig --module postfix`

The meanings of the variables are briefly commented on the configuration file `/etc/sysconfig/postfix`.

The following provides a more detailed description.

- **POSTFIX_RELAYHOST**. If the local email server should use a relay host to deliver emails that cannot be locally delivered, the relay host itself or the domain of the relay host must be given here.

If the name of a domain is provided, Postfix determines the relay host for the domain by an MX lookup.

If Postfix should forward all emails that cannot be locally delivered to a relay host without carrying out an MX lookup, the host name of the relay must be given in square brackets (for example, **[mailrelay.digitalairlines.com]**).

It is also possible to give an IP address in this form.

Optionally, the domain or host can be extended with a port number (for example, **digitalairlines.com:1025**).

If you leave this entry empty, Postfix delivers all mails that cannot be delivered locally to the mail exchanger.

Any entries in the file `/etc/postfix/transport` have precedence over the relay host.

If this variable is assigned a value, the variable `relayhost` in the file `/etc/postfix/main.cf` will be modified by running **SuSEconfig**.

- **POSTFIX_MASQUERADE_DOMAIN**. If your own DNS domain is configured with this variable (for example, `digitalairlines.com`), all addresses in emails that contain a host prefix are shortened by this host prefix.

For example, `geeko@da2.digitalairlines.com` becomes `geeko@digitalairlines.com`.

If this variable is assigned a value, the variable `masquerade_domains` in the file `/etc/postfix/main.cf` is modified by running **SuSEconfig**.

Additionally, the variable `masquerade_exceptions = root` will be set.

- **POSTFIX_LOCALDOMAINS**. Contains a comma-separated list of the domains for which Postfix should accept emails.

These values are written to the variable `mydestination` in the file `/etc/postfix/main.cf` by running **SuSEconfig**.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

If `POSTFIX_LOCALDOMAINS` is empty, the variable is set to **`$myhostname, localhost.$mydomain`** by `SUSEconfig`.

- **POSTFIX_NULLCLIENT.** A nullclient is a host that can only send mail over the network, does not receive mail over the network, and does not deliver any mail locally.

If you enter **yes**, the variable `mydestination` in the file `/etc/postfix/main.cf` will remain empty after running **SUSEconfig**.

The default entry is **no**.

- **POSTFIX_DIALUP.** If this value is set to **yes**, emails that cannot be delivered locally are not sent to their destination until the command **sendmail -q** is run.

The setting is useful for dial-up systems; otherwise, error messages would appear when sending emails if the system is not online, or a connection would be established for every email message if dial-on-demand is used.

The value **no** leads to an immediate attempt to deliver any emails waiting for delivery.

If this variable is assigned the value **yes**, the line `defer_transports = smtp` will be added to the file `/etc/postfix/main.cf` by running **SUSEconfig**.

- **POSTFIX_NODNS:** If this variable is set to **yes**, Postfix will not carry out any DNS lookups for the sender and recipient domains when processing emails.

If this variable is assigned the value **yes**, the variable `disable_dns_lookups = yes` in the file `/etc/postfix/main.cf` will be activated by running **SUSEconfig**.

- **POSTFIX_CHROOT.** If this variable is set to **yes**, the services will be run in a chroot environment, if possible. You can find the chroot environment in `/var/spool/postfix`.

If the variable is set to **no** (default), all Postfix processes will run in the normal environment.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

- **POSTFIX_UPDATE_CHROOT_JAIL.** If SuSEconfig is to set up the chroot environment, this value should be set to **yes**.
By default, the variable is set to **no**.
- **POSTFIX_LAPTOP.** Some Postfix services access FIFOs frequently, thus preventing the hard disk from spinning down.
However, if this is desired on notebooks for power-saving purposes, the variable can be set to **yes**.
- **POSTFIX_UPDATE_MAPS.** If SuSEconfig is to create the database files from the corresponding lookup tables, this variable should be set to **yes** (default).
- **POSTFIX_MAP_LIST.** If POSTFIX_UPDATE_MAPS is set to yes, you can select the lists Postfix should support here.
- **POSTFIX_RBL_HOSTS.** Here you can specify a comma-separated list of host names from which RBLs (Realtime Blackhole List) can be obtained.
No mail is accepted from clients that are these lists.
This entry makes sense only if
POSTFIX_BASIC_SPAM_PREVENTION is **not** set to **off**.
- **POSTFIX_BASIC_SPAM_PREVENTION.** Here, specify how strict filter rules for UCE (unsolicited commercial email) should be configured.
Possible levels are **off**, **medium**, and **hard**.
More details you can find at <http://www.postfix.org/uce.html>.
- **POSTFIX_MDA.** Here, specify an MDA with which Postfix should cooperate.
The entries are
 - **procmail.** Use Procmail to deliver mail locally.
 - **cyrus.** Use lmtpl to deliver to cyrus-imapd.
 - **local.** Use Postfix local MDA.

- **POSTFIX_SMTP_AUTH_***. These variables control the behavior of Postfix with respect to the authentication: if Postfix accepts mail and if Postfix delivers mail to other mail servers.
- **POSTFIX_SMTP_TLS_SERVER, POSTFIX_SMTP_TLS_CLIENT**. If these variables are set to **yes**, Postfix can encrypt the communication with the other side when sending and receiving mail, provided the following variables are configured.
- **POSTFIX_SSL_*, POSTFIX_TLS_***. These variables control various aspects of the certificate and key management needed for the encryption.

Encrypted connections are not covered in this course; this manual does not provide any details about the individual variables.

- **POSTFIX_ADD_***: These variables can be used to set the Postfix variables.

The variable must be converted to uppercase letters and appended to **POSTFIX_ADD_**.

For example, to set the Postfix variable `message_size_limit` to 100000, enter

POSTFIX_ADD_MESSAGE_SIZE_LIMIT=100000

in `/etc/sysconfig/postfix`.

Subsequently, `SuSEconfig` will generate the respective entry `message_size_limit=100000` in `/etc/postfix/main.cf`.

All available Postfix variables can be listed by using **postconf**.

- **POSTFIX_REGISTER_SLP**. If this is set to **yes**, Postfix registers automatically to SLP.

Apart from this method, further settings can be made directly in the file `/etc/postfix/main.cf`, which has very detailed comments. Following a manual modification of the file `/etc/postfix/main.cf`, modifying `/etc/sysconfig/postfix` and subsequently running of `/sbin/SuSEconfig` will not affect the file `/etc/postfix/main.cf`.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Instead, the file `/etc/postfix/main.cf.SuSEconfig` will be created, which can be renamed to `/etc/postfix/main.cf` if necessary.

Configure Postfix with `/etc/postfix/main.cf`

The main configuration file for Postfix is

`/etc/postfix/main.cf`

This file is well documented, including detailed comments.

If you decide to configure Postfix directly by editing the configuration file `/etc/postfix/main.cf`, set the variable `MAIL_CREATE_CONFIG` in `/etc/postfix/mail` to **no**.

This will prevent SuSEconfig from overwriting the configuration file.



In case there are multiple lines containing settings for variables, the settings of the last definition will be used. This allows putting all your configuration lines at the end of the configuration file.

Some important variables are the following:

- **queue_directory**. The directory in which the mail queue is located. The default entry for this is `/var/spool/postfix`.
- **command_directory**. The directory in which the Postfix administration tools are located.

The default entry is `/usr/sbin`.

- **daemon_directory**. The directory in which the Postfix daemon is located.

The default entry is `/usr/lib/postfix`.

- **mail_owner**. Describes the owner of the mail queue.

By default, this is set to **postfix**.

- **myhostname.** Defines the host name of the computer. This value serves later as the default value for other parameters.
By default, the FQDN is given here.
- **mydomain.** The domain name of the computer.
This value serves later as the default value for other parameters.
- **myorigin.** The domain that appears as the sender for emails sent locally.
The default value is the FQDN.
- **mydestination.** Describes a list of domains for which the computer should accept emails.
- **masquerade_domains.** For sender addresses of the specified domain(s), the host part is removed.
For example, geeko@da2.digitalairlines.com becomes geeko@digitalairlines.com.
- **masquerade_exceptions.** Specifies the users that should not be masqueraded. By default root is entered here.
- **relayhost.** All emails that cannot be processed locally are sent to the computer specified here.
- **inet_interfaces.** Specifies the network addresses on which Postfix waits for incoming mail.
The default value is 127.0.0.1.
To enable Postfix to receive mail from other hosts, enter the IP numbers of the network cards or **all**.
- **mynetworks.** Lists IP ranges belonging to your network.
Postfix can be configured to forward mail from hosts in these networks.

If you don't want to specify the IP ranges of your network by hand, you can use the option **mynetworks_style** which allows three values:

- ❑ **class.** Postfix trusts all SMTP clients in the same IP class (A/B/C).
- ❑ **subnet.** Postfix trusts all SMTP clients in the same IP subnet.
- ❑ **host.** Postfix trusts only the local host.
- **smtpd_recipient_restrictions, smtpd_helo_restrictions, smtpd_client_restrictions, smtpd_sender_restrictions.**
Control who is allowed to forward email over the mail server.

The variables that are relevant for most deployment scenarios are in the file

/etc/postfix/main.cf

Variables that are not defined here are assigned default values or remain empty.

To list all variables used by Postfix and their respective values, enter

postconf

Configure General Scenarios

The following scenarios presume that the variable **MAIL_CREATE_CONFIG** in the file **/etc/sysconfig/mail** is set to **no**.

If it is, the file **/etc/postfix/main.cf** will not be changed by executing **SuSEconfig**, and the file **/etc/postfix/main.cf.SuSEconfig** will not be generated.

Because these files usually contain useful settings, only few modifications are necessary for some deployment scenarios.

However, remember that the ***last*** entry of a variable in the file `/etc/postfix/main.cf` is valid.

If an entry is changed, the change does not take effect if a different value is assigned later in the file.

The following topics are described:

- Forward Mail to the Provider's Mail Server
- Receive Mail over the Internet

Forward Mail to the Provider's Mail Server

If all mail traffic is running from a mail server at the ISP, a small network merely needs a mail server that accepts the mail from the clients and passes it to the ISP's mail server.

Because the local mail server does not serve as the mail server for the company domain from the Internet, the configuration is rather simple.

Such a mail server has to

- Accept mail from the intranet clients.
- Reject mail delivered by other clients.
- Possibly rewrite sender addresses.
- Submit all mail to the provider's mail server.

Only few changes are needed in the file `/etc/postfix/main.cf`.

The following entries merely ensure that Postfix only accepts mail from the clients in the local network:

```
# 10.0.0.51 is the IP in the LAN
inet_interfaces = 10.0.0.51, 127.0.0.1
mynetworks = 10.0.0.0/24, 127.0.0.0/8
smtpd_recipient_restrictions = permit_mynetworks, reject
```

It is necessary to rewrite addresses to make sure that the sender does not appear in the form

geeko@da51.digitalairlines.com

but in the common form

geeko@digitalairlines.com

On the other hand, the host is important for messages sent to root.

Therefore, mail addressed to root should not be rewritten.

Two entries in the file `/etc/postfix/main.cf` are sufficient for this simple scenario:

```
masquerade_exceptions = root
masquerade_domains = digitalairlines.com
```

Moreover, Postfix must be informed of the mail server to which it is supposed to deliver the mail.

The relayhost entry also ensures that Postfix does not attempt to establish a direct contact to respective mail servers of the recipients.

```
relayhost = da1.digitailairlines.com
```

Exercise 6-1 Send Mail in the Local Network

In this exercise, you send mail in the local network. You configure Postfix and test your configuration.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Receive Mail over the Internet

If the mail server is set up not only for sending email messages of the users in the local network but also for receiving mail from the Internet addressed to the domain, configuring it is a bit more difficult.

It is important to prevent the mail server from being misused as an open relay by spammers.

Regardless of the individual configuration of Postfix, the server must be introduced to the DNS as the responsible mail server by means of an MX record.

In addition to the requirements in the last section, the mail server has to

- Accept mail that comes from the Internet and is addressed to your domain
- Reject mail that comes from the Internet and is not addressed to your domain
- Reject mail from known spam sources

Accordingly, a number of additional entries are needed.

As mail can theoretically be received at all interfaces, a different value is necessary for `inet_interfaces`. `mynetworks_style` can remain unchanged:

```
inet_interfaces = all
mynetworks_style = subnet
```

Postfix has to know for which domains it is can accept mail:

```
myhostname = da51.digitalairlines.com
mydomain = digitalairlines.com
mydestination = $myhostname, localhost.$mydomain,
               $mydomain
```

If Postfix is not only responsible for the mail of your domain but also for the mail of other domains (as is normally the case with web hosters), the domains are not entered under `mydestination` but in the lookup table **virtual**, which is covered in following section.

The decision to accept or not accept mail is controlled by the following variables, which contain various criteria.

- `smtpd_helo_restrictions`
- `smtpd_sender_restrictions`
- `smtpd_recipient_restrictions`
- `smtpd_client_restrictions`

A message is only delivered if it passes all the criteria without being rejected.

For example, `smtpd_sender_restrictions` can be used to prevent known spammers from delivering mail.

If the sender is listed in an RBL, the message can be rejected before the system checks whether it is addressed to a local user:

```
maps_rbl_domains = rbl-domains.digitalairlines.com
smtpd_sender_restrictions = reject_maps_rbl
```

The following entry ensures that email from the range specified in `$mynetworks` as well as email for which Postfix is responsible due to the specifications in `$mydomain` is accepted—all other mail is rejected due to `reject_unauth_destination`:

```
smtpd_recipient_restrictions = permit_mynetworks,
reject_unauth_destination
```



An explanation of all possibilities of the restrictions variables would exceed the scope of this course.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Exercise 6-2 Use Postfix on the Internet

In this exercise, you configure Postfix to send email to the Internet.

You will find this exercise in the workbook.

(End of Exercise)

Configure the Lookup Tables

Lookup tables contain rules for processing email within the overall Postfix system.

These tables are activated by variables in the file

/etc/postfix/main.cf

The tables are then defined as

/etc/postfix/lookup-table

After a lookup table has been defined, it needs to be converted to the required format (usually in the form of a hash table) using the command **postmap**.

This is done by entering:

```
postmap hash:/etc/postfix/lookup-table
```

The structure of lookup tables is subject to the following general rules:

- Blank lines or lines that begin with a # are not interpreted as command lines.
- Lines that begin with a space are regarded as a continuation of the previous line.

It is also possible to use regular expressions.

Instead of domain names, you also can use IP addresses.



A man page exists for every lookup table: **man 5 lookup-table**.

The following lookup tables are described:

- The access Lookup Table
- The canonical Lookup Table
- The recipient_canonical Lookup Table
- The sender_canonical Lookup Table
- The relocated Lookup Table
- The transport Lookup Table
- The virtual Lookup Table
- The aliases Lookup Table

The access Lookup Table

You can use the `/etc/postfix/access` lookup table to reject or allow email from defined senders.

The `smtpd` daemon evaluates this table when email arrives.

The following topics are described:

- Activate the Lookup Table
- The access Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf`:

```
smtpd_sender_restrictions = hash:/etc/postfix/access
```

The access Lookup Table Format

Each line defines a rule that is evaluated by smtpd when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of the definition of an email address in the first column and a defined action in the second column.

Possible values for email address patterns are

- ***user@domain***. Defines a filter for the specified email address.
- ***domain.name***. Defines a filter for all email addresses of the specified DNS domain.
- ***user@***. Defines a filter for all email addresses with the same user part.

Possible values for actions are

- ***4xx Text, 5xx Text***. Rejects email with the specified numerical code (see RFC821) and the defined text message.
- **REJECT**. Rejects the email with a generic error message.
- **OK**. Accepts the email.
- **DISCARD *optional text***. Makes sure that the email is discarded without an error message to the sender.

The ***optional text*** appears in the log file. If no text is specified, a generic message appears in the log.

Examples:

```
postmaster@digitalairlines.com  OK
spam@hahaha.net                550 We're fighting against spam!
194.95.93.10                    REJECT
```




See the man pages (**man 5 access**) for other possible actions.

The canonical Lookup Table

You can use the `/etc/postfix/canonical` lookup table to rewrite sender and recipient addresses of incoming and outgoing emails.

Both the header and the envelope are rewritten.

The cleanup daemon reads this table when an email arrives.

The following is described:

- Activate the Lookup Table
- The canonical Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf`:

```
canonical_maps = hash:/etc/postfix/canonical
```

The canonical Lookup Table Format

Each line defines a rule that is evaluated by `smtpd` when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of the definition of an email address in the first column and a defined action in the second column.

Possible values for email address patterns are

- ***user@domain***. Defines a filter for the specified email address.

- ***user***. Defines a filter for all email addresses with the same user part, provided the domain part of the email is listed in one of the variables \$myorigin, \$mydestination, \$inet_interfaces, or \$proxy_interfaces in the /etc/postfix/main.cf file.
- ***@domain***. Defines a filter for all email addresses of the specified domain.

Possible values for action are

- ***user@domain***. Rewrites the email address to the value defined here.

Examples:

```
training@digitalairlines.com    geeko@digitalairlines.com
@slc.digitalairlines.com       slc@digitalairlines.com
```

If you want to convert sender addresses and recipient addresses in a different way, use

- **recipient_canonical** to convert the recipient addresses
- **sender_canonical** to convert the sender addresses

The recipient_canonical Lookup Table

You can use the /etc/postfix/recipient_canonical lookup table to convert recipient addresses of incoming and outgoing emails.

The cleanup daemon evaluates this table when an email arrives before the generic lookup table /etc/postfix/canonical is evaluated.

The following topics are described:

- Activate the Lookup Table
- The recipient_canonical Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf` by the entry

```
recipient_canonical_maps = hash:/etc/postfix/recipient_canonical
```

The recipient_canonical Lookup Table Format

Each line defines a rule that is evaluated by `smtpd` when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of the definition of an email address in the first column and a defined action in the second column.

Possible values for email address patterns are

- ***user@domain***. Defines a filter for the specified email address.
- ***user***. Defines a filter for all email addresses with the same user part, provided the domain part of the email is listed in one of the variables `$myorigin`, `$mydestination`, `$inet_interfaces`, or `$proxy_interfaces` of the file `/etc/postfix/main.cf`.
- ***@domain***. Defines a filter for all email addresses of the specified domain.

Possible values for actions are

- ***user@domain***. Rewrites the email addresses to the value defined here.

Examples:

```
geeko@digitalairlines.com      training@digitalairlines.com  
@slc.digitalairlines.com      slc@digitalairlines.com
```

The sender_canonical Lookup Table

You can use the `/etc/postfix/sender_canonical` lookup table to rewrite sender addresses of incoming and outgoing emails (for outgoing email: *login@host.internal.com* to *firstname.surname@mycompany.com*).

The cleanup daemon reads this table when an email arrives before the generic lookup table `/etc/postfix/canonical` is read.

The following topics are described:

- Activate the Lookup Table
- The sender_canonical Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf` by the entry

```
sender_canonical_maps = hash:/etc/postfix/sender_canonical
```

The sender_canonical Lookup Table Format

Each line defines a rule that is evaluated by `smtpd` when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of the definition of an email address in the first column and a defined action in the second column.

Possible values for email address patterns are

- *user@domain*. Defines a filter for the specified email address.

- ***user***. Defines a filter for all email addresses with the same user part, provided the domain part of the email is listed in one of the variables \$myorigin, \$mydestination, \$inet_interfaces, or \$proxy_interfaces of the file /etc/postfix/main.cf.
- ***@domain***. Defines a filter for all email addresses of the specified domain.

Possible values for actions are

- ***user@domain***. Rewrites the email address to the value defined here.

Examples:

```
training@digitalairlines.com    geeko@digitalairlines.com
@slc.digitalairlines.com       slc@digitalairlines.com
```

The relocated Lookup Table

You can use the /etc/postfix/relocated lookup table to return the corresponding bounced email, with a note of the new address of the desired addressee, to senders of emails to users that no longer exist on this system.

The following topics are described:

- Activate the Lookup Table
- The relocated Lookup Table Format

Activate the Lookup Table

This function is activated in the file /etc/postfix/main.cf by the entry

```
relocated_maps = hash:/etc/postfix/relocated
```

The relocated Lookup Table Format

Each line defines a rule that is evaluated by smtpd when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of a key field in the first column, which refers to the email address of the former recipient or defines this by means of a regular expression and contact information in the second column, which may contain a new email address of the recipient or other contact information.

Possible values for the key field are

- ***user@domain***. Defines a filter for the specified email address.
- ***user***. Defines a filter for all email addresses with the same user part, provided the domain part of the email is listed in one of the variables \$myorigin, \$mydestination, \$inet_interfaces, or \$proxy_interfaces of the file /etc/postfix/main.cf.
- ***@domain***. Defines a filter for all email addresses of the specified domain.

Possible values for contact information include any information (such as email address or telephone number) that will help someone reach the email addressee. The information is used in "user has moved to ***new_location***" bounce messages.

Examples:

```
geeko@digitalairlines.com    geeko@novell.com
tux@digitalairlines.com     Please call 1-800-PIRATES
```

The notifications of the mail server are sent by email to the sender:

```
<geeko@digitalairlines.com>: host da51.digitalairlines.com[10.0.0.51]
said: 550
    <geeko@digitalairlines.com>: Recipient address rejected: User has
moved to
    geeko@novell.com (in reply to RCPT TO command)

<tux@digitalairlines.com>: host da51.digitalairlines.com[10.0.0.51] said:
550
    <tux@digitalairlines.com>: Recipient address rejected: User has moved
to
    Please call 1-800-PIRATES (in reply to RCPT TO command)
```

The transport Lookup Table

You can use the `/etc/postfix/transport` lookup table to define email routing for special email address ranges.

The following is described:

- Activate the Lookup Table
- The transport Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf` by the entry

```
transport_maps = hash:/etc/postfix/transport
```

The transport Lookup Table Format

Each line defines a rule that is evaluated via the `qmgr` or the `trivial-rewrite` daemon before an email is sent.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line consists of the definition of a domain pattern in the first column and a defined transport path in the second column.

Possible values for the domain pattern are

- ***user@domain***. Email to the specified user is forwarded over the defined transport route.
- ***domain***. All email to the specified domains are forwarded via the defined transport path.
- ***.domain***. All email with subdomains under the specified domain are forwarded via the defined transport path. This is only important if `transport_maps` is not listed in the variable `parent_domain_matches_subdomain`; otherwise, ***domain*** also includes ***.domain***.

Possible values for the transport path are

- ***transport:nexthop***. Different values can be assigned to ***transport***, such as ***local***, ***smtp***, or ***uucp***. Also, any transport path can be assigned to ***transport***, including self-defined paths (such as Cyrus and Procmail).
 - ***local***. Defines the delivery of email via the Postfix process `local` that delivers the email in the local system.
For this specification, the value for ***:nexthop*** remains blank.
 - ***smtp***. Defines the delivery of email via the Postfix process `smtp`, which delivers the email to a remote mail exchanger via the SMTP protocol.
host or ***host:port*** can be configured as ***nexthop*** for an email exchanger on a remote host in case it does not accept email on port 25/TCP.
To prevent DNS lookups on the MX entry, the form ***[host]*** or ***[host]:port*** should be used for the ***nexthop*** entry.

- ❑ **uucp.** Defines the delivery of email via the Postfix process pipe, which is configured by means of the file `/etc/postfix/master.cf` for the delivery of email via UUCP.

The recipient host is specified as *nexthop*.

Examples:

```
digitalairlines.com    smtp:da51.digitalairlines.com:10025
suse.com               uucp:da150
```

The virtual Lookup Table

You can use the `/etc/postfix/virtual` lookup table to set up email for a number of domains with separate user names.

The following topics are described:

- Activate the Lookup Table
- The virtual Lookup Table Format

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf` by the entry

```
virtual_maps = hash:/etc/postfix/virtual
```

The virtual Lookup Table Format

Each line defines a rule that is evaluated via `smtpd` when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Using virtual domains requires the definition of the virtual domain first. This is done by placing the virtual domain name in the first column and arbitrary text in the second column. This text is only used to keep the structure of the file and has no meaning.

Every other line describing a recipient address of this domain contains

- First column: The recipient address.
- Second column:
 - The user name of the local email user to whom the incoming email should be delivered.
 - or
 - A comma-separated list of all local email users to whom incoming emails should be delivered.

When you specify a virtual domain, only email addresses containing this virtual domain are modified. Address with a subdomain or host name are not modified. You need to specify them as virtual domains first.

Example:

virtual.domain	geeko, tux
postmaster@virtual.domain	postmaster
user1@virtual.domain	geeko
user2@virtual.domain	tux

The aliases Lookup Table

The /etc/aliases lookup table is used to define aliases. You cannot redirect emails to mailboxes on other hosts or domains.

The following topics are described:

- Activate the Lookup Table
- The aliases Lookup Table Format

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Activate the Lookup Table

This function is activated in the file `/etc/postfix/main.cf` by the entry

```
alias_maps = hash:/etc/aliases
```

The aliases Lookup Table Format

Each line defines a rule that is evaluated by `smtpd` when an email arrives.

The rules are processed from top to bottom and the matching of rules ends when the first match occurs.

Each line contains

- First column: A local recipient address followed by a colon.
- Second column: Filtered email is then redirected to another email user or to another email alias.

Details of the target recipient in the second column can also be extended to include multiple recipients using a comma-separated list.

An email is delivered explicitly to a local user if the recipient address in the second column begins with a “\”.

The following is an example:.

```
root:          \root, geeko
mailer-daemon: root
postmaster:    mailer-daemon
daemon:        root
webmaster:     tux@digitalairlines.com
wwwrun:        webmaster
```

If the file `/etc/aliases` has been modified, it must be converted into the hash table `/etc/aliases.db` by entering

```
da51:~ # postalias /etc/aliases
```

or

```
da51:~ # newaliases
```

Exercise 6-3 Use Lookup Tables

In this exercise, you use the Postfix lookup tables.

You will find this exercise in the workbook.

(End of Exercise)

Use Postfix Tools

Apart from the previously mentioned tools, Postfix also has a whole range of other useful administration tools that can make life considerably easier for a postmaster.

This section briefly introduces the administration tools for Postfix:

- **newaliases.** Converts the ASCII file `/etc/aliases` to the hash table `/etc/aliases.db`.
- **mailq.** Lists all emails in the mail queues that have not yet been sent.
- **postalias.** Converts the ASCII file `/etc/aliases` to the hash table `/etc/aliases.db`. Same as **newaliases**.
- **postcat.** Displays the contents of a file from the queue directories in a readable form.
- **postconf.** Without any parameters, this tool displays the values of all variables defined in the file `/etc/postfix/main.cf` as well as the values used by the standard variables. To modify variables directly, enter

postconf -e *key=value*

These changes are automatically integrated in the file `main.cf`.

- **postdrop.** This is run automatically by using the **sendmail** command, if **sendmail** cannot write any files to the maildrop directory because of missing world-writable permissions. It saves the forwarded email as `sgid maildrop`.
- **postfix.** Enables configuration errors to be found (**postfix check**), forces email from the deferred queue to be delivered immediately (**postfix flush**), or rereads the Postfix configuration files (**postfix reload**).
- **postmap.** Generates the hash tables for the lookup tables in the directory `/etc/postfix/`.

- **postsuper**. Checks the file structure in the queue directories and removes unneeded files and directories (**postsuper -s**) or deletes files and directories that have been left after a system crash and are useless (**postsuper -p**).

Individual email messages can be removed from the mail queues with **postsuper -d ID**.

In general, **postsuper** removes all files that are not normal files or directories (such as symbolic links).



Run the command **postsuper -s** immediately before starting the Postfix system.



For more information about these tools, see the man page **man 1 Postfix-Tool**.

Objective 2 **Apache Web Server**

To set up a Apache web server on SUSE Linux Enterprise Server, you need to know the following:

- Setup a Basic Web Server
- Configure Virtual Hosts

Other configuration tasks, like limiting access to the web server with .htaccess files, PHP, etc., are specific to Apache and do not differ substantially between RHEL 4 and SLES 10.

Setup a Basic Web Server

- The Basic Functionality of a Web Server
- Install a Basic Apache Web Server
- Understand the Structure and the Basic Elements of the Apache Configuration Files
- Understand the Default Apache Configuration

The Basic Functionality of a Web Server

A web server delivers data that is requested by a web browser. The data can have different formats such as HTML files, image files, Flash animations, or sound files.

Web browsers and web servers communicate using HTTP (Hyper Text Transfer Protocol).

In addition to delivering data to the web browser, a web server can perform tasks such as limiting access to specific web pages, logging access to a file, and encrypting the connection between a server and browser.

Install a Basic Apache Web Server

To set up a basic Apache web server, you need to do the following:

- Install the Required Software Packages
- Start and Test the Web Server
- Locate the DocumentRoot of the Web Server

Install the Required Software Packages

To run a basic Apache web server, you need to install the following packages with YaST:

- **apache2.** The basic web server software.
- **apache2-prefork.** An additional Apache package that influences the multiprocessing behavior of the web server.
- **apache2-example-pages.** Sample HTML pages.

When you install the packages listed above, YaST prompts you to also install one or more additional packages required by Apache. Confirm the additional package installation by selecting **OK** to resolve all dependencies of the Apache packages.

Start and Test the Web Server

After installing the required software, you need to start the web server. Do this as the root user by entering the following:

rcapache2 start

As with all services, enter the following to stop the web server:

rcapache2 stop

If you want the web server to start up at boot time, you need to enter the following:

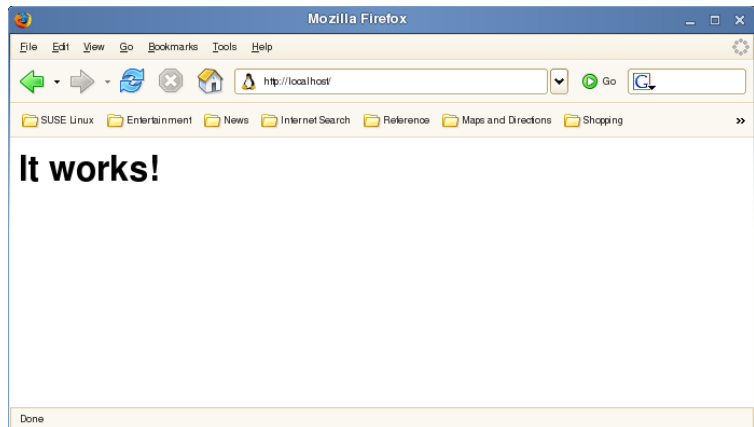
insserv apache2

To test whether the web server is properly installed, open a web browser and enter the following address:

http://localhost/

The browser displays the following page:

Figure 6-4



If your SUSE Linux Enterprise Server 10 is connected to a network, you (and other hosts on the network) can remotely access the web server by entering the following:

http://your_system_IP_address/

If your network provides a DNS server, you can use the hostname instead of the IP address.

Locate the DocumentRoot of the Web Server

The default directory of the data provided by Apache is **`/srv/www/htdocs/`**.

This directory is also called the *DocumentRoot* of the web server. After the installation, it contains the Apache example pages, which are displayed above.

You can replace the data in the DocumentRoot directory to display your own web server content. Because the web server runs with the user id `wwwrun`, you have to make sure that this user has *read* access to files in the DocumentRoot directory.

If you create subdirectories in DocumentRoot, you can access those subdirectories with the following web address scheme:

`http://your_server/name_of_subdirectory`

If no specific file is requested in the address, Apache looks for a file with the name `index.html`. You can change the name of this default file in the Apache configuration files.

Exercise 6-4 Install Apache

In this exercise, you install the apache components on your system

You can find the exercise in the workbook.

(End of Exercise)

Exercise 6-5 Test the Apache Installation

In this exercise, you check if the installation of apache was successful.

You can find the exercise, in the workbook.

(End of Exercise)

Understand the Structure and the Basic Elements of the Apache Configuration Files

To configure the Apache web server, you need to do the following:

- Locate the Apache Configuration Files
- Understand the Basic Rules of the Configuration Files

Locate the Apache Configuration Files

The configuration of the Apache web server is spread over several configuration files located in the directory `/etc/apache2/`.

The following is a list of the most important Apache configuration files:

- **httpd.conf.** This is the main Apache configuration file.
- **default-server.conf.** This file contains the basic web server setup. However, all options set in this file can be overwritten by other configuration files.
- **vhost.d/.** This is a directory containing configuration files for virtual host setups. You will learn more about virtual hosts later in this section.
- **uid.conf.** This configuration file sets the user and group id for Apache. By default, Apache uses the user id `wwwrun` and the group id `www`.
- **listen.conf.** In this configuration file, you can specify the IP addresses and TCP/IP ports Apache is listening to. By default, Apache listens to all assigned interfaces on port 80.
- **server-tuning.conf.** You can use this configuration file to fine tune the performance of Apache. The default values should be fine unless you are going to run a web server that has to handle a lot of requests at the same time.
- **error.conf.** In this file you configure the behavior of Apache when a request cannot be performed correctly.

- **ssl-global.conf.** Configure the connection encryption with SSL in this configuration file.

Understand the Basic Rules of the Configuration Files

The options of the Apache configuration files are called *directives*. Directives are case sensitive, which means that a word such as “include” is not the same as “Include.”

Directives can be grouped so that they do not apply to the global server configuration. In the following, the directives only apply to the directory /srv/www/htdocs:

```
<Directory "/srv/www/htdocs">
    Options None
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

The directives are grouped by <Directory “/srv/www/htdocs”> and </Directory> which limits their validity to the directory /srv/www/htdocs only.

You can use the # character to indicate comments in the configuration file. All lines starting with a # are ignored by the Apache server.

Whenever you edit the Apache configuration files, you need to reload the web server by entering the following:

rcapache2 reload

In some cases it is not enough to reload Apache. You need to stop and restart the web server by entering the following:

rcapache2 restart

If you are not sure that your changes use the correct syntax, you can verify the syntax of the configuration files by entering the following:

apache2ctl configtest

If the syntax is correct, the command displays the following message:

```
Syntax OK
```

Understand the Default Apache Configuration

The main Apache web server configuration is defined in the file **/etc/apache2/default-server.conf**. The following is an overview of the most important directives used in that file:

Table 6-1

Directive	Description
DocumentRoot	Specifies the DocumentRoot of the web server.
<Directory “dir_name”> </Directory>	All directives used within this block apply only to the specified directory.
Options	With this directive additional options can be applied to logical blocks like directories.
AllowOverride	Determines whether directives are allowed to be overwritten by a configuration found in a .htaccess file in a directory.
Alias “fakename” “realname”	Allows you to create an alias to a directory.

*(continued)***Table 6-1**

Directive	Description
ScriptAlias	Allows you to create an alias to a directory containing scripts for dynamic content generation.

In most cases the default settings are suitable and don't need to be changed.



An overview of all Apache directives can be found at <http://httpd.apache.org/docs-2.0/mod/directives.html>.

Configure Virtual Hosts

To use the virtual host feature of Apache, you need to know the following:

- The Concept of Virtual Hosts
- Configure a Virtual Host

The Concept of Virtual Hosts

With the default setup, the Apache server can be reached with a browser using the following web addresses (URLs):

- **`http://localhost`** (from the computer where the web server is running)
- **`http://web_server_IP_address`**
- **`http://web_server_hostname`**

For all of these addresses, Apache serves the same files located in the DocumentRoot directory.

Using this setup, you would need a dedicated computer for every domain of the Internet. To avoid this, Apache can be configured to host multiple virtual web servers on one physical system. These virtual web servers are called virtual hosts.

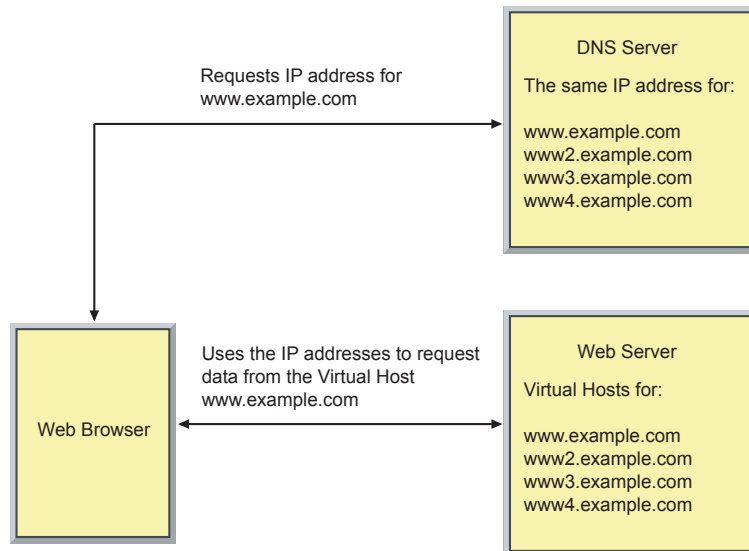
To access virtual hosts, a DNS entry is needed for every virtual host of the Apache web server.

The following outlines the steps of sending a request to the virtual host `www.example.com`:

1. The web browser requests the IP address of the host `www.example.com`.
2. The browser uses the IP address to request a file from the Apache web server listening on the IP address of `www.example.com`.
3. In the HTTP request, the browser includes the hostname of the server it wants to reach.
4. Apache uses the hostname to determine the corresponding virtual host and delivers the requested data from that host.

The following illustrates this process:

Figure 6-5



Configure a Virtual Host

For every virtual host you need to create a configuration file in the directory `/etc/apache2/vhosts.d/`. The name of the configuration file has to end with `.conf`.

You can find a template file `vhost.template` in the directory `/etc/apache2/vhosts.d/` to use as a base for your configuration file.

You need to edit the following directives in the template:

Table 6-2

Directive	Description
ServerAdmin	Enter the email address of the Virtual Host administrator here.

(continued) **Table 6-2**

Directive	Description
ServerName	Enter the hostname of the virtual host as it is configured in the DNS.
DocumentRoot	Set the DocumentRoot of the virtual host. The directory and the files in the directory must be readable by the user wwwrun.
ErrorLog	Enter a filename for the error log. The file must be writable for the user wwwrun.
CustomLog	Enter a filename for the general log file. The file must be writable for the user wwwrun.
ScriptAlias	Set the ScriptAlias to a directory of your choice. The directory must not be under the DocumentRoot of the virtual host. If you don't need scripts for dynamic content creation, delete this directive.
<Directory "script_dir">	If you've set a ScriptAlias before, you have to configure a directory which contains the script files. If you are not using a ScriptAlias, delete this directory block.
<Directory "document_root">	You need to adjust the path name of this directory directive to the path of your DocumentRoot.

After customizing the template file, you need to reload the Apache web server. You also need to make sure that the settings in DNS are updated so that the **hostname** of your virtual host is resolved correctly.

Exercise 6-6 Configure a Virtual Host

In this exercise, you configure a virtual host for the accounting department.

You can find this exercise, in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Summary

Objective	Summary
1. Postfix	<p>The Postfix Mail Delivery Agent consists of several modules that process different queues designed to receive and send mail.</p> <p>The Postfix master daemon is configured via the file /etc/postfix/master.cf</p> <p>All other configuration parameters, with the exception of the lookup tables, are contained in /etc/postfix/main.cf.</p> <p>On SUSE Linux Enterprise Server 10, the most common parameters of this file can be modified using variables in the files /etc/sysconfig/mail and /etc/sysconfig/postfix</p> <p>Modifications in the file /etc/sysconfig/postfix are only adopted in the file /etc/postfix/main.cf and, in some cases, in the file /etc/postfix/master.cf after executing /sbin/SuSEconfig.</p> <p>Lookup tables contain rules for processing email within the overall Postfix system.</p>

Objective	Summary
1. Postfix (contd.)	<p>The following are Postfix tools:</p> <ul style="list-style-type: none"> ■ newaliases. Converts the ASCII file <code>/etc/aliases</code> to the hash table <code>/etc/aliases.db</code>. ■ mailq. Lists all email in the mail queues that have not yet been sent. ■ postalias. Converts the ASCII file <code>/etc/aliases</code> to the hash table <code>/etc/aliases.db</code>. Same as newaliases. ■ postcat. Displays the contents of a file from the queue directories in a readable form. ■ postconf. Displays the values of all variables. Enter postconf -e key=value to modify variables directly. These changes are automatically integrated in the file <code>main.cf</code>. ■ postdrop. This is run automatically by the command <code>sendmail</code>. ■ postfix. Enables configuration errors to be found, forces email from the deferred queue to be delivered immediately, or rereads the Postfix configuration files. ■ postmap. Generates the hash tables for the lookup tables in the directory <code>/etc/postfix/</code>. ■ postsuper. Removes all files that are not normal files or directories.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
2. Apache Web Server	<p>Apache is the leading web server software.</p> <p>For a basic web server, you need to install the following packages:</p> <ul style="list-style-type: none">■ apache2■ apache2-prefork■ apache2-example-pages <p>The locally running web server can be accessed using the address http://localhost/.</p> <p>The default document root of the web server is <code>/etc/www/htdocs</code>.</p> <p>The Apache configuration files are located in the directory <code>/etc/apache2</code>.</p> <p>The options of the Apache configuration files are called directives.</p> <p>You can check the syntax of the configuration file with the command apache2ctl configtest.</p> <p>By configuring virtual hosts you can host multiple domains on one physical machine.</p> <p>You need to create a configuration file in the directory <code>/etc/apache2/vhosts.d/</code> for every virtual host.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 7 AppArmor

Both, RHEL 4 and SUSE Linux Enterprise Server 10, provide tools to implement Mandatory Access Control (MAC). RHEL 4 uses SELinux, while SUSE Linux Enterprise Server 10 uses AppArmor. SELinux and AppArmor share a similar purpose and both use the Linux Security Modules to implement MAC. However, beyond that they are distinctly different and there are no tools to convert the SELinux configuration to an AppArmor configuration.

In this section, you will learn to configure AppArmor to protect services running on SUSE Linux Enterprise Server 10.

Objectives

1. Understand the Difference between SELinux and AppArmor
2. Create and Manage AppArmor Profiles
3. Control AppArmor
4. Monitor AppArmor

Objective 1 Understand the Difference between SELinux and AppArmor

- Discretionary Access Control
- Linux Security Modules
- Mandatory Access Control

Discretionary Access Control

With Discretionary Access Control (DAC) the user (subject) has complete control over the files (objects) he owns or the programs he executes.

DAC describes which files a process can access based on the UID of the process. In general, the UID of the process is the same as the UID of the user who started the process. An exception are binaries where the SUID bit is set.

One of the main problems with DAC Process security is that if a process is hacked or is out of control for some reason, this process can access all the files which are owned by the process' UID as well as the files belonging to other UIDs which it may access according to the file permissions set.

The command chroot is a way to restrict a process from accessing selected files. However, the chroot command is known for its limitations and restrictions.

Linux Security Modules

To solve this problem, the Linux Security Modules (LSM) were developed. In an LSM enabled system a process (Subject) can only access the files (Objects) defined in a policy. This policy is set by the system administrator and enforced by the operating system kernel; access to files is no longer at the discretion of the user.

According to the kernel documentation, the LSM kernel patch provides a general kernel framework to support security modules. In particular, the LSM framework is primarily focused on supporting access control modules, although future development is likely to address other security needs such as auditing. By itself, the framework does not provide any additional security; it merely provides the infrastructure to support security modules.

The best known Linux tools based on LSM are Security Enhanced Linux (SELinux, used in RHEL) and AppArmor (used in SLES). With these tools you can define with high granularity which process can access which file with which permissions. These rules are enforced by the kernel and are independent of the UID of the process and the permissions set in the file system.

To define these policies you must have a thorough understanding of the respective applications and the way they work.

This kind of access control is also known as Mandatory Access Control (MAC).

Mandatory Access Control

In contrast to Discretionary Access Control, where the user determines the permissions granted to the object she creates, Mandatory Access Control permissions are set by the administrator according to a policy.

The following is described:

- SELinux
- AppArmor

SELinux

Mandatory access control in SELinux is possible in two ways: Type Enforcement (TE) and Role Based Access Control (RBAC). To learn more about these, we recommend the web site of the National Security Agency at <http://www.nsa.gov/selinux/>. The NSA initiated the SELinux project.

The main configuration files for SELinux on an RHEL system are stored in the directory **/etc/selinux/**.

SELinux is normally activated by the init process at the boot time. Activation during boot can be disabled by entering the boot parameter **selinux=0** or by setting the **SELINUX** parameter in **/etc/selinux/config** to **disabled**.

There are two types of SELinux policies: targeted (only targeted network daemons are protected) and strict (full SELinux protection). Only one type of policy can be active in the Kernel at a time. Which policy is used when activating SELinux is defined by the parameter **SELINUXTYPE** in the file **/etc/selinux/config**.

Policies are created using the m4 macro language. The sources are kept in **/etc/selinux/<POLICY-NAME>/src/policy/**, while the binaries can be found in **/etc/selinux/<POLICY-NAME>/policy/**.

Policies are managed by the Security-Server which is a core SELinux component. Messages from the Security-Server are stored in the `/var/log/messages`.

Using the command **ls -Z** you can view the security context used for a file. Using the **-Z** option in combination with the **ps** command shows the security context of a running process.

The command **setenforce** is used to switch SELinux from “on” to “warn only” and vice versa. **system-config-securitylevel** is a tool that allows to change SELinux configuration parameters. The command **selinuxenabled** can be used to check the status of SELinux on a running system. The return code 0 signifies that SELinux is enabled.

AppArmor

The AppArmor kernel modules (`apparmor` and `aamatch_pcre`) hook into the Linux Security Modules Framework of the kernel.

Profiles in `/etc/apparmor.d/` are used to configure which application may access and execute which files.

AppArmor has to be activated before the applications it controls are started. Applications already running when AppArmor is activated are not controlled by AppArmor, even if a profile has been created for them. Therefore, AppArmor is activated early in the boot process by `/etc/init.d/boot.apparmor`.

In a default installation of SUSE Linux Enterprise Server 10, AppArmor is actively protecting several services using a set of profiles provided by Novell. Using the provided YaST modules or command line tools, you can easily adapt these to your needs and create new profiles for additional applications you want to protect.

As a general rule, you should confine programs that grant privilege, i.e., programs that have access to resources that the person using the program does not have:

- Network agents—programs that have open network ports
- Cron jobs
- Web applications



At some points in the documentation of AppArmor you will find AppArmor's former name, Subdomain.

AppArmor protection can be turned on and off using the command **rcapparmor** start and stop, respectively. Single applications can be toggled between **enforce** (AppArmor limits what an application may do) and **complain** (only log entries are created) mode with commands of the same name.

Objective 2 Create and Manage AppArmor Profiles

SUSE Linux Enterprise Server 10 comes with AppArmor Profiles for various applications, such as named, ntpd, nsd, and others.

The profiles are contained in **/etc/apparmor.d/**. The filename of the profile represents the filename of the application including the path, with “/” being replaced by a “.”: The profile for `/usr/sbin/squid` would be contained in `/etc/apparmor.d/usr.sbin.squid`.

A profile can include other files with an **#include** statement. The directory **/etc/apparmor/abstractions/** contains several files that are intended to be included in profiles, depending on the kind of program to be protected by AppArmor. There are abstractions for files that should be readable or writable by all programs (base), for name service-related files like `/etc/passwd` (nameservice), for files related to console operations (console), and others.

The profiles are plain text files and it is therefore possible to create and modify them with any text editor. However, command line tools as well as a YaST module greatly simplify the process of creating profiles.

In addition to the active profiles in `/etc/apparmor.d/`, several profiles are already prepared in `/etc/apparmor/profiles/extras/` that you can customize to your needs and copy to `/etc/apparmor.d/` to activate them.

To successfully administer AppArmor, you need to

- Understand Profiles and Rules
- Administer AppArmor Profiles with YaST
- Administer AppArmor Profiles with Command Line Tools

Understand Profiles and Rules

Novell AppArmor **profiles** contain two types of AppArmor **rules**: path entries and capability entries. Path entries specify what a process can access in the file system. AppArmor, by default, limits the capabilities a process is given (see `man apparmor`). Capability entries are used to specify specific POSIX capabilities (`man 7 capabilities`) a process is granted, overriding the default limitation.

Other files containing AppArmor rules can be pulled in with **#include** statements.

As an example, let's have a look at the profile for `/sbin/klogd`, the kernel log daemon (`/etc/apparmor.d/sbin.klogd`):

```
1 # Profile for /sbin/klogd
   #include <tunables/global>
5 /sbin/klogd {
   #include <abstractions/base>

   capability sys_admin,

10  /boot/System.map*      r,
   /proc/kmsg             r,
   /sbin/klogd            rmix,
   /var/log/boot.msg      rwl,
   /var/run/klogd.pid     rwl,
15 }
```

Comments (as in line1) start with a # sign,

#include (as in line 3 and 6) is not interpreted as a comment, but is used to include rules from other files. The path as given above is relative to **/etc/apparmor.d/**.

/etc/apparmor.d/tunables/global (line 3) is used to include definitions that should be available in every profile. By default, it just includes **/etc/apparmor.d/tunables/home**, which defines the variables **@{HOMEDIRS}** and **@{HOME}**. These variables are used in various profiles.

The directory **/etc/apparmor.d/abstractions/** contains files with general rules grouped by common application tasks. These include, for instance, access to files all applications need (base), access to authentication mechanisms (authentication), graphics environments (kde, gnome), name resolution (nameservice), and others. Instead of having these redundantly specified in several profiles, they are defined at one point and included in the profiles that need them.

Line 5 in the example above gives the absolute path to the program confined by AppArmor. The rules as well as any includes follow within the curly braces {}.

Line 8 enables the capability **sys_admin** for this program. Any other capabilities needed would be listed in separate lines starting with **capability**.

The remaining lines list files and directories, and the access permission granted.

Within lines listing files and directories, the following wildcards can be used:

- *. Substitutes any number of characters, except /.
- **. Substitutes any number of characters, including /. Use ** to include subdirectories.
- ?. Substitutes any single character, except /.
- [abc]. Substitutes a, b, or c.
- [a-d]. Substitutes a, b, c, or d.
- {ab,cd}. Substitutes either ab or cd.

The permissions granted can be

- **r.** Allows the program to have read access to the resource. Read access is required for scripts, and an executing process needs this permission to allow it to dump core or to be attached to with `ptrace`.
- **w.** Allows the program to have write access to the resource. Files must have this permission if they are to be unlinked (removed).
- **l.** Link mode mediates access to symlinks and hardlinks and grants the privilege to unlink (remove) files.
- **m.** Allow executable mapping. This mode allows a file to be mapped into memory using `mmap(2)`'s `PROT_EXEC` flag. This flag marks the pages executable; it is used on some architectures to provide non-executable data pages, which can complicate exploit attempts. AppArmor uses this mode to limit which files a well-behaved program (or all programs on architectures that enforce non-executable memory access controls) may use as libraries, to limit the effect of invalid `-L` flags given to `ld(1)` and `LD_PRELOAD`, `LD_LIBRARY_PATH` given to `ld.so(8)`.
- **ix.** Inherit Execute Mode. The executed resource inherits the current profile.
- **px.** Discrete Profile Execute Mode. This mode requires that a profile be defined for the resource executed. If there is no profile defined, access is denied.
- **Px.** Discrete Profile execute mode -- scrub the environment. **Px** allows the named program to run in **px** mode, but AppArmor will invoke the Linux Kernel's `unsafe_exec` routines to scrub the environment, similar to `setuid` programs. (See `man 8 ld.so` for some information on `setuid/setgid` environment scrubbing.)

1 **HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED**

- **ux.** Unconstrained Execute Mode. Allows the program to execute the resource without any Novell AppArmor profile being applied to the executed resource. ***This should only be used in rare exceptions.***
- **Ux.** Unconstrained execute -- scrub the environment. As ux, it should only be used in rare exceptions.

The last five, ix, px, Px, ux, and Ux, cannot be combined.

The manual page covering the syntax of the profiles is
man 5 apparmor.d

Administer AppArmor Profiles with YaST

The profile for /sbin/klogd, given in the example above, is a rather short profile. When you browse through the profiles in **/etc/apparmor.d/** or **/etc/apparmor/profiles/extras/** you will see that these profiles can be much more complex.

AppArmor comes with several tools that help to create and maintain AppArmor profiles. YaST modules exist that provide a graphical interface to those tools.

You can accomplish various tasks with YaST:

- Create a New Profile
- Update a Profile
- Delete a Profile

Create a New Profile

To create a new profile, you can

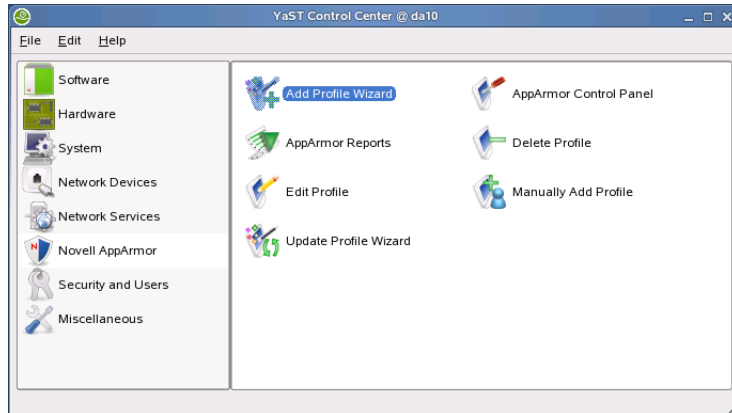
- Use the New Profile Wizard
- Manually Create a New Profile

Use the New Profile Wizard

There is a Wizard to create a new profile. Before calling the Wizard to profile an application, the first step is to stop the application you want to create a profile for.

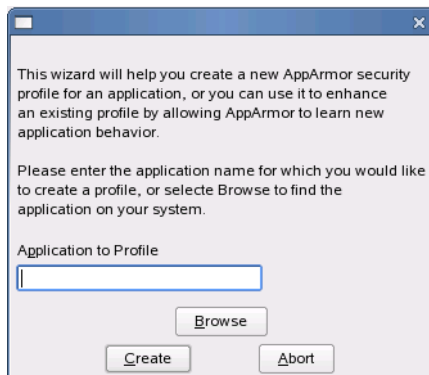
To access the Add Profile Wizard, start YaST and select Novell AppArmor:

Figure 7-1



Then select the Add Profile Wizard. The first step is to enter the application you want to profile:

Figure 7-2



If no path is given, the Wizard looks for the binary in the search path (variable \$PATH).

The next dialog asks you to start and use the application you want to profile. Use the application in a way that you expect it to be used in production. For instance, if you are profiling a web server, access it in a way that you expect it to be accessed during normal operation. For a web browser, use it in a way you expect the users to access web content.

During this learning phase, any access to files or capabilities needed by the application is granted as well as logged in the log file **/var/log/audit/audit.log**. Because any access is granted, you have to make sure that no attack can happen during this phase of profile creation. AppArmor does not yet protect your application.

Once you feel you have gone through all expected uses of the application, select **Scan system log for AppArmor events** in the YaST AppArmor Profile Wizard dialog.

For each event you are presented with a dialog where you can decide what should happen when this event occurs in the future. The dialog offers different options, depending on the event.

In case of access to a program, the dialog looks similar to the following:

Figure 7-3



- **Inherit.** The executed resource inherits the current (parent's) profile.
- **Profile.** Requires that a specific profile exist for the executed program.
- **Unconfined.** Executes the program without a security profile. Do not run unconfined unless absolutely necessary.
- **Deny.** The execution of the program will be denied.

In case of file access, the dialog offers different options:

Figure 7-4



The Add Profile Wizard suggests an access mode (r, w, l, or a combination of them). If more than one item appears in the list of files, directories, or #includes, select the radio button in front of the appropriate one, and then select one of these buttons:

- **Allow.** Grants the program access to the specified directory path.
- **Deny.** Prevents the program from accessing the specified file or directory.

Sometimes the suggested files or directories do not fit your needs. In this case, you can modify them:

- **Glob.** Selecting Glob once replaces the filename with an asterisk, including all files in the directory. Selecting Glob twice replaces the file and the directory it resides in by **, including all directories and files of that level in the directory tree. Selecting Glob again goes up one level in the path.
- **Glob w/Ext.** Selecting Glob w/Ext once replaces the filename with an *, but retains the file name extension: text.txt becomes *.txt. Selecting Glob w/Ext twice replaces the file and the directory it resides in by **, retaining the file name extension: /a/b/c/text.txt becomes /a/b/**/*.txt.
- **Edit.** Enables editing of the highlighted line. The new (edited) line appears at the bottom of the list.

After you have modified the line, select Allow or Deny. Go through each learning mode entry in this way.

Once all entries have been processed, you are returned to the AppArmor Profile Wizard dialog that asked you to run the application. You can run the application again, and then run through any additional entries generated by this.

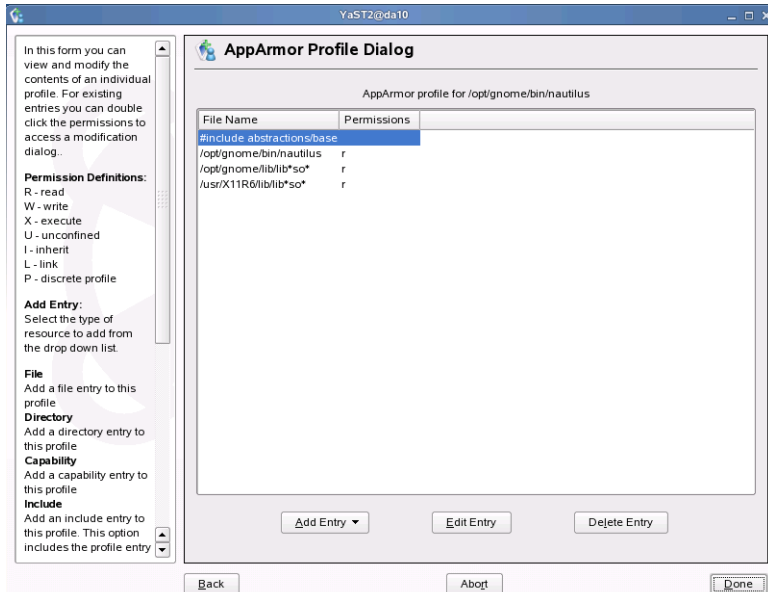
Once you conclude you are done, select **Finish**. The profile is written and activated.

If you want to discard your selections, select **Abort**.

Manually Create a New Profile

When selecting **YaST > Novell AppArmor > Manually Add Profile**, you are prompted to select a file for which you want to create a profile. Subsequently the AppArmor Profile Dialog opens. There you can Add, Edit, and Delete entries to the profile by selecting the respective buttons.

Figure 7-5



The advantage of the YaST module is its syntax check. However, you can also use any text editor, like vi, to create and edit profiles.

Update a Profile

To update a profile, you have two choices:

- Run the Add Profile Wizard Again
- Run the Update Profile Wizard

Run the Add Profile Wizard Again

When you run the Add Profile Wizard on a program for which a profile already exists, the profiling does not start from scratch, but uses the existing profile as a basis.

You go through the same steps as if generating a new profile, but most likely you will have to answer fewer questions than on the first run.

This method is suitable to update a specific profile, especially for client applications that run a finite amount of time.

Run the Update Profile Wizard

When you want to update several profiles, or profiles for applications that run over a longer period of time, using the Update Profile Wizard is the better choice.

Even though the Update Profile Wizard is a YaST module, you may need to take some preparatory steps with command line tools.

The first step is to decide which application profiles you want to update, and to put AppArmor into complain (also called learning) mode with regard to these applications. (If there is no profile yet for an application you want to profile, you have to create one first, using **autodep program**.)

The command **complain** is used to activate learning or complain mode. You can either use the program or the profile as the argument: For instance, both, **complain firefox** and **complain /etc/apparmor.d/usr.lib.firefox.firefox.sh** work to change AppArmor to learning mode for Firefox.

If you want to turn on learning mode for all applications confined by AppArmor, use **complain /etc/apparmor.d/***.

In profiles that are in complain mode, the path to the application being confined is followed by **flags=(complain)**:

```
# Profile for /sbin/klogd

#include <tunables/global>

/sbin/klogd flags=(complain){
...
}
```

Then actually use your application(s) to create events in the log file.

The next step is to start the Update Profile Wizard by starting YaST and selecting **Novell AppArmor > Update Profile Wizard**:

Figure 7-6



The interface is almost identical to the Add Profile Wizard interface; also the choices you are presented do not differ.

However, as you are updating different profiles, you have to pay special attention to the profile in the first line to be sure that your decision on allowing or denying fits the respective profile.

Once the log file has been processed, select **Finish**. The profiles will be reloaded, but AppArmor is still in complain mode.

To have AppArmor again enforce the rules, use the command `enforce`, which has the same syntax as `complain`:
enforce /etc/apparmor.d/* puts all profiles in enforce mode.

Delete a Profile

To delete a profile, start YaST and select **Novell AppArmor > Delete Profile**. Select the profile to delete; then select Next. After you select **Yes** in the confirmation dialog, the profile is deleted and the application is no longer confined by AppArmor.

Administer AppArmor Profiles with Command Line Tools

There are various tools to create and maintain AppArmor profiles. These are

- autodep
- genprof
- logprof
- vim

autodep

autodep generates a profile skeleton for a program and loads into the Novell AppArmor module in complain mode.

The syntax is **autodep *program1 program2 ...***

genprof

genprof (Generate Profile) is used to create a profile for an application. Stop the application you want to create a profile for before running genprof.

genprof runs autodep on the specified program if there is no profile yet, puts the new or already existing profile in complain mode, marks the log file, and prompts the user to start the program to profile and to exercise its functionality.

```
da10:~ # genprof firefox
Setting /usr/lib/firefox/firefox.sh to complain mode.

Please start the application to be profiled in
another window and exercise its functionality now.

Once completed, select the "Scan" button below in
order to scan the system logs for AppArmor events.

For each AppArmor event, you will be given the
opportunity to choose whether the access should be
allowed or denied.

Profiling: /usr/lib/firefox/firefox.sh

[(S)can system log for SubDomain events] / (F)inish
```

Once the user has done that and presses **s** in the terminal window where genprof is running, genprof calls logprof to run against the system log from where it was marked when genprof was started.

In case of access to a program, the dialog looks similar to the following:

```
Reading log entries from /var/log/audit/audit.log.
Updating subdomain profiles in /etc/apparmor.d.

Profile: /usr/lib/firefox/firefox.sh
Program: firefox
Execute: /bin/basename
Severity: unknown

[(I)nherit] / (P)rofile / (U)nconfined / (D)eny / Abo(r)t /
(F)inish
```

In case of access to a file or directory, the dialog looks similar to this:

```
Complain-mode changes:

Profile:  /usr/lib/firefox/firefox.sh
Path:    /dev/tty
Mode:    rw
Severity: 9

  1 - #include <abstractions/consoles>
  [2 - /dev/tty]

[(A)llow] / (D)eny / (G)lob / Glob w/(E)xt / (N)ew /
Abo(r)t / (F)inish
```

Press the number to switch lines as applicable, then press the letter in parenthesis corresponding to what you want to do. The options offered are the same as those within YaST; **New** corresponds to a certain extent to **edit** in YaST.

Once all log entries have been processed, you are returned to `genprof`, where you can start a new scan or Finish the profile generation.

```
Writing updated profile for /usr/lib/firefox/firefox.sh.

Profiling: /usr/lib/firefox/firefox.sh

[(S)can system log for SubDomain events] / (F)inish
```

logprof

`logprof` is a tool used to scan the log `/var/log/audit/audit.log` for entries created by AppArmor for profiles in learning mode, and to interactively create new profile entries.

The choices you have are the same as those described under “`genprof`” on 7-21.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

If you want logprof to start scanning from a certain point in the log file, you can pass a string that describes that point. The following is an entry in the log file:

```
type=APPARMOR msg=audit(852099290.789:1103): PERMITTING w
access to /root/.mozilla/firefox/uwgzf7zy.default/prefs.js
(firefox-bin(8770) profile /usr/lib/firefox/firefox.sh
active /usr/lib/firefox/firefox.sh)
```

Using the option **-m**: **logprof -m "852099290.789:1103"**, you can start the scan of the log file from that point, ignoring earlier entries.

vim

The profiles can be changed using any text editor.

Compared to other editors, vim has the advantage that AppArmor includes a syntax highlighting description that enables vim to highlight syntax elements in profiles.



When confining Apache2, you can create subprofiles (also called hats) that use a different security context, for instance for pages using mod_php5. To make use of such subprofiles, an application has to be made “hat-aware.” For Apache2 this is achieved with module mod_change_hat that comes with AppArmor on SLES10. For details on hats, see **man change_hat** and section 5 of the **Administration Guide** in /usr/share/doc/manual/apparmor-admin_en/apparmor-admin_en.pdf. Subprofiles are part of the profile for the application itself and are administered with the same tools (genprof, logprof).

Exercise 7-1 AppArmor

In this exercise you create, test, and improve a profile for the Firefox browser.

You will find this exercise in the workbook.

(End of Exercise)

Objective 3 **Control AppArmor**

AppArmor can be controlled using the script **/etc/init.d/boot.apparmor** or the link to this script, **/sbin/rcapparmor**. This script takes the usual parameters start, stop, etc, but because AppArmor is not a daemon, their significance is slightly different.

To control AppArmor, you have to know how to

- Start and Stop AppArmor
- View AppArmor's Status
- Reload Profiles

Start and Stop AppArmor

To confine an application, AppArmor has to be active before the application starts. Therefore, AppArmor is usually activated early in the boot process.

If you do not want AppArmor to confine your applications any longer, use **rcapparmor stop**. This unloads the profiles, but the AppArmor kernel modules `apparmor` and `aamatch_pcre` remain loaded. **rcapparmor kill** unloads the kernel modules as well. In both cases, applications are no longer confined.

rcapparmor start activates AppArmor. However, only applications started after the activation of AppArmor are confined. Even if, for instance, a profile for Squid exists, Squid will not be confined if it was already running before you started AppArmor. To include Squid in AppArmor's protection, you need to restart Squid after activating AppArmor.

View AppArmor's Status

rcapparmor status gives you a general overview of profiles and processes:

```
da10:~ # rcapparmor status
apparmor module is loaded.
50 profiles are loaded.
49 profiles are in enforce mode.
1 profiles are in complain mode.
Out of 69 processes running:
5 processes have profiles defined.
5 processes have profiles in enforce mode.
0 processes have profiles in complain mode.
```

To emphasize the point that, after restarting AppArmor, processes need to be restarted to be again confined, have a look at the following:

```
da10:~ # rcapparmor stop
Unloading AppArmor profiles           done
da10:~ # rcapparmor start
Loading AppArmor profiles             done
da10:~ # rcapparmor status
apparmor module is loaded.
50 profiles are loaded.
49 profiles are in enforce mode.
1 profiles are in complain mode.
Out of 62 processes running:
0 processes have profiles defined.
0 processes have profiles in enforce mode.
0 processes have profiles in complain mode.
```

Restarting one of the processes for which there is a profile changes the output of **rcapparmor status**:

```
da10:~ # rcnscd restart
Shutting down Name Service Cache Daemon           done
Starting Name Service Cache Daemon                 done
da10:~ # rcapparmor status
apparmor module is loaded.
50 profiles are loaded.
49 profiles are in enforce mode.
1 profiles are in complain mode.
Out of 62 processes running:
1 processes have profiles defined.
1 processes have profiles in enforce mode.
0 processes have profiles in complain mode.
```

The output of AppArmor does not contain specific data regarding the profiles or the processes being confined.

A list of the profiles loaded is kept in **/sys/kernel/security/apparmor/profiles**. It might look like the following:

```
da10:~ # cat /sys/kernel/security/apparmor/profiles
/usr/sbin/traceroute (enforce)
/usr/sbin/squid (enforce)
/usr/sbin/sendmail (enforce)
/usr/sbin/postqueue (enforce)
...
/usr/lib/postfix/bounce (enforce)
/usr/lib/firefox/firefox.sh (complain)
/usr/bin/ldd (enforce)
...
```

The command **unconfined** lists processes that have bound sockets but have no profiles loaded:

```
da10:~ # unconfined
2659 /sbin/portmap not confined
2659 /sbin/portmap not confined
2694 /usr/lib/zmd/zmd-bin not confined
2756 /usr/sbin/slpd not confined
2756 /usr/sbin/slpd not confined
2756 /usr/sbin/slpd not confined
2756 /usr/sbin/slpd not confined
2756 /usr/sbin/slpd not confined
2756 /usr/sbin/slpd not confined
2831 /usr/sbin/cupsd not confined
2831 /usr/sbin/cupsd not confined
2874 /usr/sbin/sshd not confined
2905 /usr/sbin/sshd not confined
2905 /usr/sbin/sshd not confined
3040 /usr/lib/postfix/master not confined
3040 /usr/lib/postfix/master not confined
```

This does not give information about processes with profiles that are not confined because they were running already when AppArmor was activated. To spot those, have a look at **ps -Z** and compare the output with the content of `/sys/kernel/security/profiles`; restart any processes that should be confined.

Reload Profiles

If you have changed profiles in `/etc/apparmor.d/` manually with an editor (not by using the AppArmor tools like `logprof`), you have to reload the profile or profiles concerned. The command to use is **rcapparmor reload**. **rcapparmor restart** is equivalent to `reload`; it does not stop and then start AppArmor, but it does reload the profiles. Processes that were confined before **rcapparmor reload** was issued remain confined (unless you deleted their profile or changed their status from `enforce` to `complain`).

The commands **enforce** and **complain** toggle the status from enforce to complain and vice versa, and reload the profiles concerned.

Objective 4 Monitor AppArmor

There are two ways to monitor AppArmor:

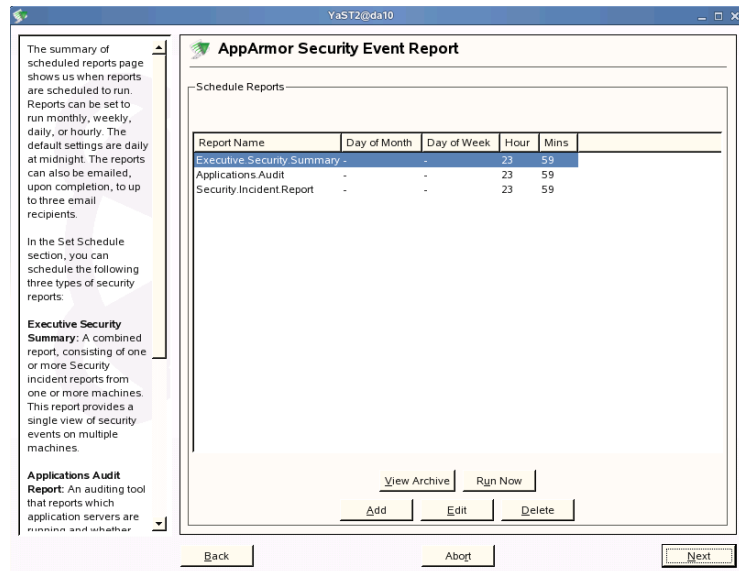
- Security Event Report
- Security Event Notification

Security Event Report

The YaST module to configure and view AppArmor security event reports can be launched by starting YaST and selecting **Novell AppArmor > AppArmor Reports**. It can also be launched directly from a console window as root by entering **yast2 SD_Report**.

The dialog that opens up shows when security event reports are generated:

Figure 7-7



By default these are created once a day at midnight.

Using the buttons **Add**, **Edit**, or **Delete**, you can schedule new security incident reports, edit the existing ones, for instance, to set the email address that should receive the report, or delete event reports.

Selecting a report and then selecting **Run Now** either shows the result directly, or, in the case of the Security Incident Report, first opens a dialog where you can fine tune the content of the resulting report:

Figure 7-8

The Report Configuration dialog enables you to filter the report selected in the previous screen. To filter by **Date Range**:

1. Click **Filter By Date Range**. The fields become active.
2. Enter the start and end dates that delineate the scope of the report.
3. Enter other filtering parameters. See below for definitions of parameters.

The following definitions help you to enter the filtering parameters in the Report Configuration Dialog: **Program Name Pattern**: When you enter a program name or pattern that matches the name of the executable process of interest, the report will display security events

Report Configuration Dialog

☐ **Filter By Date Range**

Select Date Range

Enter Starting Date/Time

Hours: [0] Minutes: [0] Day: [1] Month: [1] Year: [2005]

Enter Ending Date

Hours: [0] Minutes: [0] Day: [1] Month: [1] Year: [2005]

Program name: [] Profile name: [] PID number: [] Severity: [All]

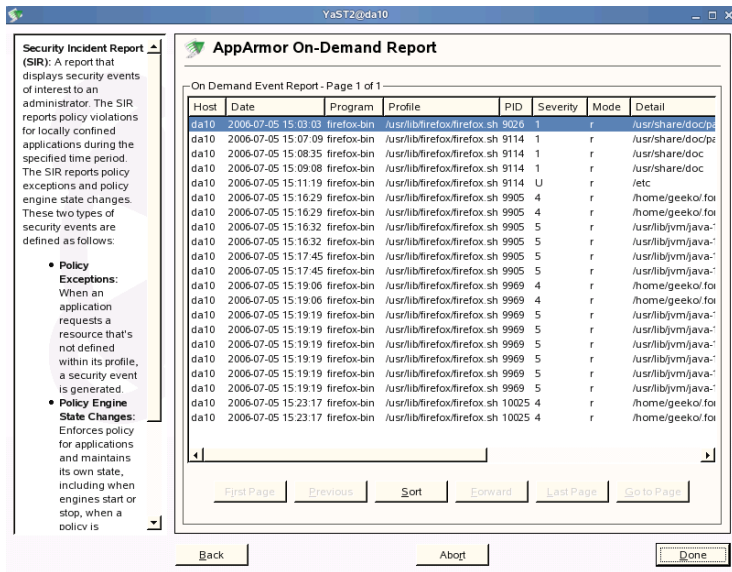
Detail: [] Access Type: [R] Mode: [All]

Export Type: [None] Location to store log: [/var/log/apparmor/reports-exported] [Browse]

[Back] [Abort] [Next]

The help text on the left explains the available options. Once you configured what you want to have included in your report, select **Next**. The report is displayed, showing the security events:

Figure 7-9



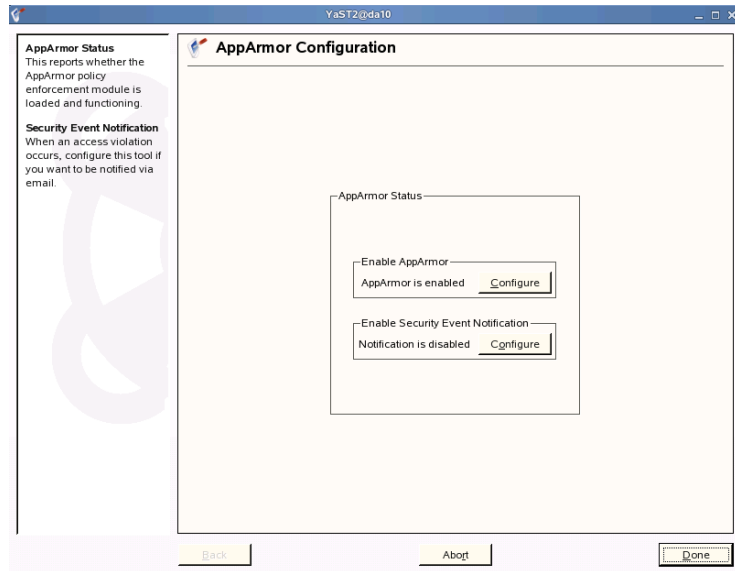
Depending on your configuration in the previous dialog, it is also saved to disk, by default in the directory `/var/log/apparmor/reports-exported/`.

Security Event Notification

To configure the security event notification, start **YaST** and select **Novell AppArmor > AppArmor Control Panel**, or start the module directly from a console window as root by entering **yast2 subdomain**.

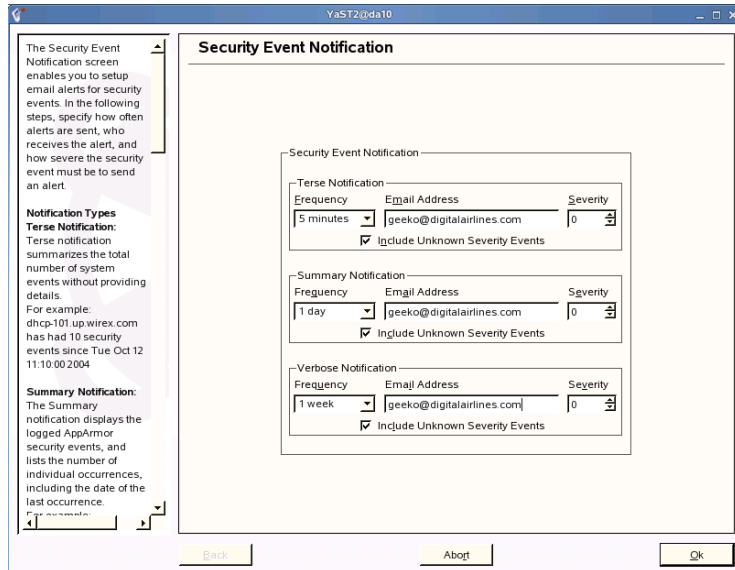
The following dialog opens up:

Figure 7-10



Select **Configure** in the **Enable Security Event Notification** box. A dialog opens up where you can configure the frequency of the notifications, email addresses, and the severity levels the reports should cover:

Figure 7-11



Select **Ok** to save your configuration. Close the AppArmor configuration window by selecting **Done**.

Summary

Objective	Summary
1. Understand the Difference between SELinux and AppArmor	<p>SELinux and AppArmor impose limits on processes based on rules set in policies and profiles.</p> <p>Even if compromised, programs are limited in what they are allowed to read, write, and execute. This even holds true for programs running with root permissions.</p> <p>While SELinux and AppArmor share a common goal, the technical implementation is entirely different.</p>
2. Create and Manage AppArmor Profiles	<p>There are YaST modules as well as command line tools to create and manage AppArmor profiles.</p> <p>The main YaST modules are the Add Profile Wizard, and the Update Profile Wizard.</p> <p>The corresponding command line tools are autodep, genprof, and logprof.</p> <p>Profiles are text files located in <code>/etc/apparmor.d/</code> and can be maintained with any text editor.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
3. Control AppArmor	<p>AppArmor is started early during the boot process.</p> <p>/etc/init.d/boot.apparmor is the script used to control AppArmor. /sbin/rcapparmor is a link to it.</p> <p>The parameters stop and kill end the confinement of processes by AppArmor. start is used to activate confinement of processes—but only processes with profiles that are started after AppArmor has been activated are confined. reload or restart reload the profiles.</p> <p>enforce and complain toggle enforce and complain mode.</p> <p>unconfined lists processes with bound sockets that have no profile.</p> <p>ps -Z lists security related information.</p>
4. Monitor AppArmor	<p>AppArmor can be configured via YaST to create reports on security events and to send email messages to inform on security events.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 8 Manage Virtualization with Xen

Virtualization is one of the hottest topics in the industry at the moment. However, the idea of virtualization is not new at all. Hardware platforms like IBMs pSeries or zSeries support virtualization since a long time and software like VMware Workstation for x86 based systems has been available for many years.

Now virtualization moves to mainstream, because affordable Intel or AMD based x86 systems, provide enough resources to run more than one virtual machine at the same time.

SUSE Linux Enterprise Server 10 is the first enterprise level Linux product that comes with build-in virtualization support through the Xen virtual machine monitor. In the following section you'll learn how to use this powerful feature.

Objectives

1. Understand the Concept of Virtualization
2. Understand How Xen Works
3. Install Xen
4. Manage Xen Domains with YaST
5. Manage Xen Domains at the Command Line
6. Understand Xen Networking
7. Migrate a Guest Domain

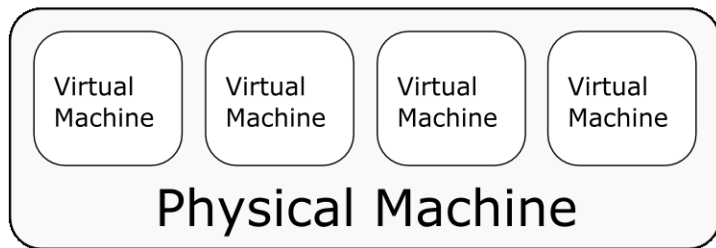
1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 1 Understand the Concept of Virtualization

Virtualization technology separates a running instance of an operating system from the physical hardware. Instead of a physical machine, the operating system runs in a so-called virtual machine. Multiple virtual machines share the resources of the underlying hardware.

Virtualization allows you to run multiple virtual systems on one single physical machine.

Figure 8-1



The following are the main advantages of virtualization, in comparison with non-virtualized physical hardware:

1. **Efficient Hardware Utilization.** Often systems are not using the full potential of their hardware. By running multiple virtual machines on the same hardware, the resources are used more efficiently.
2. **Reduced Downtime.** Virtual machines can be easily migrated to a new physical host system. This reduces the downtime in case of a hardware failure.
3. **Flexible Resource Allocation.** Hardware resources can be allocated on demand. When the resource requirements of a virtual machine change, resource allocation can be adjusted or the machine can be migrated to a different physical host.

Objective 2 Understand How Xen Works

The idea of virtualization is not new. Platforms like IBM zSeries or pSeries offer built-in virtualization and Intel x86 based systems can be virtualized using third-party software like VMware.

SUSE Linux Enterprise Server 10 comes with a virtualization technology called Xen, which allows you to run multiple virtual machines on a single piece of Intel x86 based hardware.

At the moment, the operating systems that run in a Xen virtual machine need to be modified. Therefore only open source operating systems like Linux or BSD can be installed. One exception is Netware, which has been adjusted by Novell to run in a Xen virtual machine.

Intel and AMD are developing extensions (Intel Vanderpool and AMD Pacifica) to the x86 Standard to support virtualization. Once these extensions are available, Xen will be able to run unmodified operating systems including Microsoft Windows.



You can find updated information about Xen, including an instruction how to run unmodified operating systems on the OpenSUSE Xen page at: <http://en.opensuse.org/Xen>

To understand how Xen works, you need to do the following:

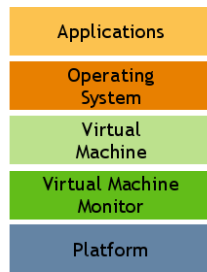
- Understand Virtualization Methods
- Understand the Xen Architecture

Understand Virtualization Methods

Before we talk in detail about the Xen technology, you should understand the following two different virtualization methods.

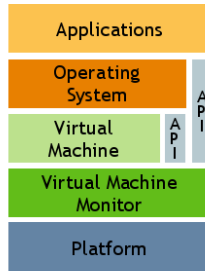
- **Full Virtualization.** In this case the virtualization software emulates a full virtual machine including all hardware resources. The operating system running in the virtual machine (guest OS) communicates with these resources as if they were physical hardware. VMware Workstation is a popular full virtualization software.

Figure 8-2



Full-virtualization

- **Para Virtualization.** Instead of emulating a full virtual machine, para-virtualization software provides an Application Programming Interface (API) which is used by the guest OS to access hardware resources. This requires that the guest OS is aware that it runs in a virtual machine and needs to know how to access the API. Xen is a para-virtualization software.

Figure 8-3**Para-virtualization**

Para virtualization provides better performance because it does not emulate all hardware details. The drawback is that the guest OS needs to be modified to run with para-virtualization. Full-virtualization works with an unmodified guest OS but generates more overhead resulting in a weaker performance.

Another advantage of para-virtualization is the flexible resource allocation. As the guest OS is aware of the virtual environment, Xen can, for example, change the memory allocation of a virtual machine on the fly without any reboot.

Understand the Xen Architecture

Xen consists of the following two major components:

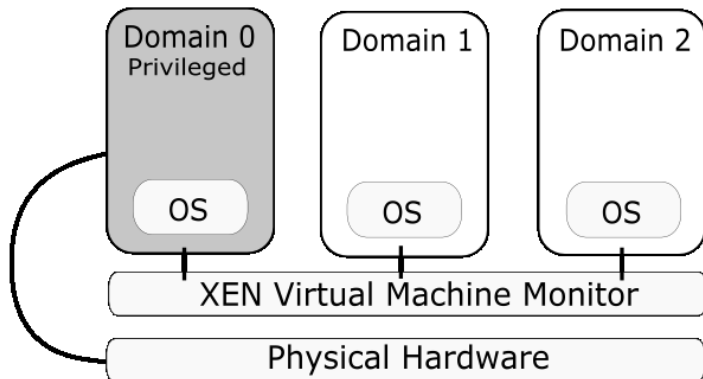
- **Virtual Machine Monitor.** The virtual machine monitor forms a layer between physical hardware and virtual machines. In general this kind of software is called a **Hypervisor**.
- **Xen tools.** The Xen tools are a set of command line applications that are used to administer virtual machines.

The virtual machine monitor must be loaded before any of the virtual machines are started. When working with Xen, virtual machines are called **domains**.

The Xen virtual machine monitor neither includes any drivers to access the physical hardware of the host machine nor an interface to communicate directly with an administrator. These tasks are performed by an operating system running in the privileged **domain0**.

The following is an overview of a Xen system with three domains.

Figure 8-4



A process called **xend** runs in the domain0 Linux installation. This process is used to manage all Xen domains running on a system and to provide access to their consoles.

A unprivileged domain is also called domainU in the Xen terminology.

SUSE Linux Enterprise Server 10 can be used for privileged (domain0) and unprivileged (domainU) Xen domains.

Objective 3 Install Xen

To set up a Xen system, you start from a normal SUSE Linux Enterprise 10 installation, which is going to run in domain0.

The other Xen domains can later be installed in physical partitions or file system images. When you plan to use physical partitions, you have to make sure that the initial SUSE Linux Enterprise Server 10 installation is not using all of the available disc space.

For maximum flexibility it makes sense to use the logical volume manager LVM or EVMS for a Xen system.

The following packages have to be installed in the initial SUSE Linux Enterprise Server 10 installation:

- **xen.** This package contains the Xen virtual machine monitor (Hypervisor).
- **xen-tools.** Contains xend and a collection of command line tools to administer a Xen system.
- **kernel-xen.** This package contains a modified Linux kernel that runs in a Xen domain.
- **xen-doc-*** (optional). Xen documentation in various formats.

The installation of the Xen package automatically adds an entry like the following into the boot loader configuration file **/boot/grub/menu.lst**.

```
title Xen
root (hd0,3)
kernel /boot/xen.gz
module /boot/vmlinuz-Xen root=/dev/hda3 selinux=0
module /boot/initrd-Xen
```



On some Xen systems you might see the parameter **dom0_mem** in the kernel module line. This parameter assigns a certain amount of initial main memory to domain0 at boot time. However in Xen version 3, this parameter is not required anymore.

Initially all available memory is used by domain0. When you start additional domainUs, the required amount of memory is reduced in domain0 and used for the new domainU.

The entry in menu.lst adds a new option to the boot menu of your system. When selecting this entry, the Xen virtual machine monitor is loaded (**kernel /boot/xen.gz**) which starts SUSE Linux Enterprise Server 10 in domain0 (see the lines starting with **module**).

Before rebooting your system with the Xen option, you should check if the automatically generated entry is correct. Make sure that ...

- ... the line **root (hd0,3)** points to the file system which contains the Xen Virtual Machine Monitor and the Kernel of the Linux installation for domain0. In our example **hd0,3** means the fourth partition on the first hard drive in the system. Also check if the parameter **root** in the first module line points to the root partition of the domain0 installation.
- ... the Xen version of the Linux kernel and the initrd are loaded in the module line. The names of the image files should end in **-xen**.

After checking the boot loader configuration file, you can reboot your system and select the Xen option at the boot loader menu. In the early stages of the boot process, you will see some messages of the Xen virtual machine monitor on the screen. Then the domain0 Linux installation is started.

In case the system is not booting properly, you can switch back to a non-virtualized system by selecting the regular SUSE Linux Enterprise Server 10 boot option.



When running Xen, the network setup is done by the xend management process. This can interfere with the native network configuration scripts of the domains. Especially SuSEfirewall2 is known to cause problems. It is therefore recommended to stop SuSEfirewall2 with **rcSuSEfirewall2** and to remove the firewall scripts from the init process:

```
insserv -r SuSEfirewall2_setup  
insserv -r SuSEfirewall2_init  
insserv -r SuSEfirewall2_final (conditional)
```

Exercise 8-1 *Install Xen*

In this exercise, you learn how to install Xen and configure domain0.

You can find this exercise in the workbook.

(End of Exercise)

Objective 4 Manage Xen Domains with YaST

After you have installed Xen and the Xen tools, you can start to create more Xen domains. Before we go into the details of the domain configuration, we will introduce the YaST module **Virtual Machine Management (Xen)**.

This module provides a convenient way to create and control the Xen domains on your system. The module can be started from the **System** section in the YaST Control Center, and has to run on the Linux system running in domain0.

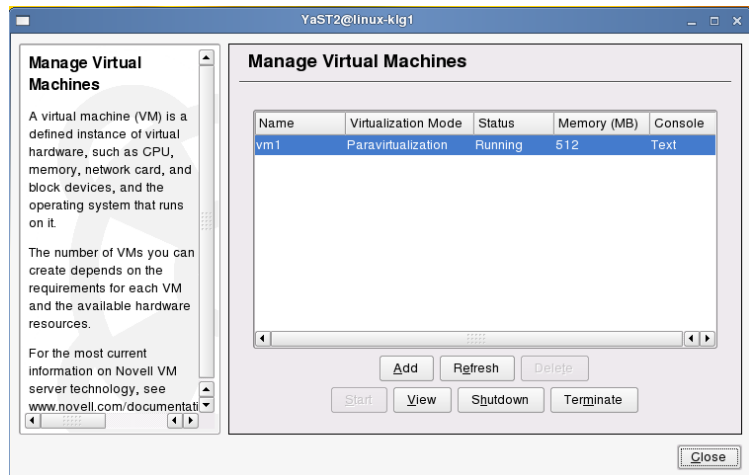


Not every detail of the Xen domain configuration is described in the following. More in-depth information follow in the next objective.

The following is a step by step description of how to create and boot a new Xen domain with this YaST module.

After you have started the module, the following dialog appears on the screen:

Figure 8-5



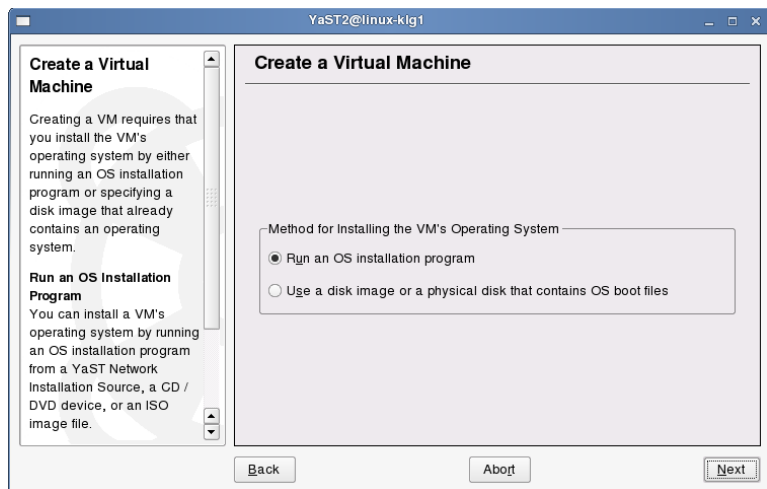
In our example there is already one guest domain running on the system, which is listed in the upper part of the dialog. The columns of the table display various information about the domain including the name, the status and the memory allocation.

The following buttons are in the lower part of the dialog:

- **Add.** Select this button to create a new domain.
- **Refresh.** This button refreshes the information about the domains.
- **Delete.** Deletes a domain completely.
- **Start.** Starts a domain.
- **View.** Opens a terminal window to access the console of a domain.
- **Shutdown.** Performs a regular shutdown of the guest OS.
- **Terminate.** Terminates the domain immediately without waiting for the guest OS to shutdown.

After selecting **Add**, the following dialog appears:

Figure 8-6

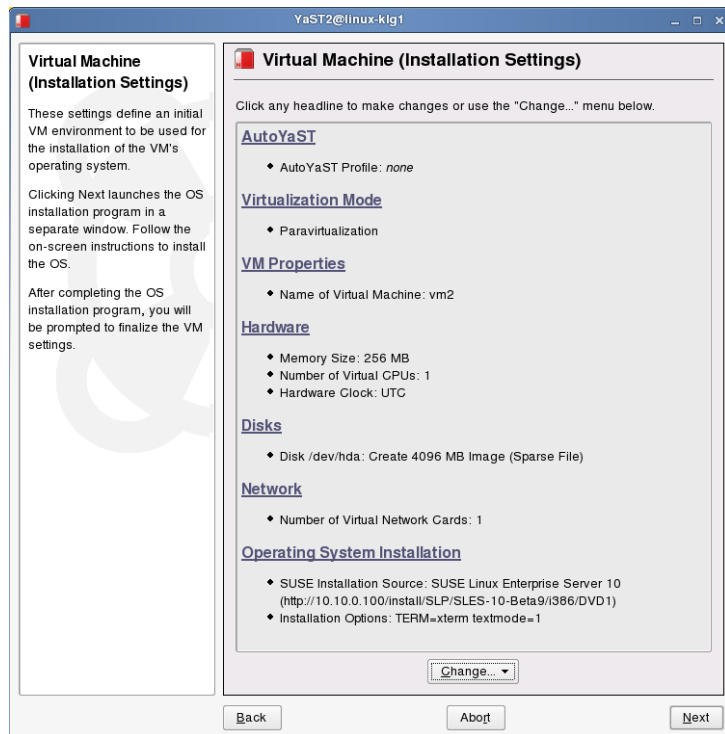


The dialog gives you two choices:

- **Run an OS installation program.** This allows you to run a SUSE Linux Enterprise Server installation from an installation source that is registered in the system.
- **Use a disk image or a physical disk that contains OS boot files.** This option lets you create a Xen domain from an existing installation in a physical disc or disc image.

For the following example we select the **Run an OS installation program** option. The following dialog appears:

Figure 8-7

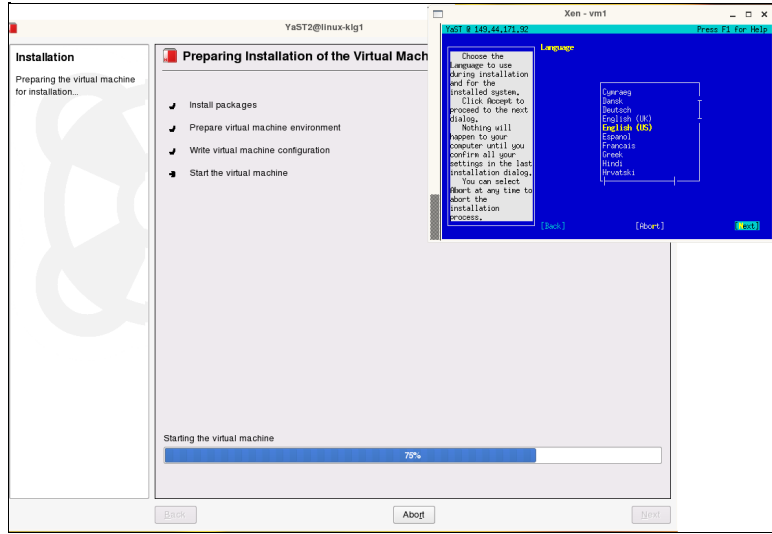


The following options can be adjusted by selecting their headlines:

- **AutoYaST.** In this option you can specify an AutoYaST profile that should be used for the installation. When there is no AutoYaST profile, a manual installation is started.
- **Virtualization.** You can switch between para virtualization and full virtualization. Full-virtualization is only available on supported hardware with Intel or AMD virtualization extension.
- **VM Properties.** Here you can change the name of the new domain.
- **Hardware.** In this option you can configure the hardware configuration of the domain. (Memory, Number of CPUs, ...)
- **Disks.** Configure the Disks here. These can either be physical block devices or file system / disc images.
- **Network.** This option lets you add additional network adapters to the domain.
- **Operating System Installation.** Here you can configure the installation source and additional installation options.

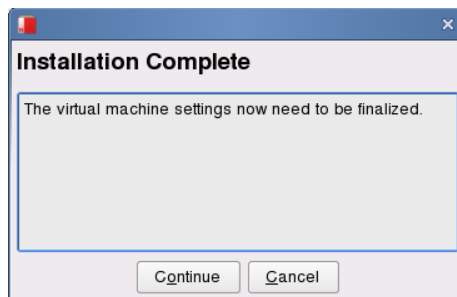
For our example we stay with the default and select **Next**. Now the domain environment and the installation process is started.

Figure 8-8



The installation itself is a standard SUSE Linux Enterprise Server installation, except that it runs in text mode. After the packages have been installed, the following dialog appears:

Figure 8-9



Select **Continue**.

The following dialog gives you a resume about the domain configuration. Usually there is nothing to do here. Select **Next** in this dialog and in the domain overview.

A terminal window opens up where you can finish the remaining steps of the OS installation with YaST.

Exercise 8-2 Install a Guest Domain

In the following you can practice how to install a Xen guest domain using YaST. Before you start with this exercise you must have installed Xen on your system.

You can find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 5 **Manage Xen Domains at the Command Line**

In the following you learn how to manage Xen domains at the command line. This includes:

- Understand a Domain Configuration File
- Use the xm Tool
- Automate Domain Startup and Shutdown

Understand a Domain Configuration File

Every Xen domain needs a configuration file. For domains which have been created with YaST, the configuration file is usually located in **/etc/xen/vm/**.

Under **/etc/xen/examples**, you find two example files, which can be used if you would like to create a configuration from scratch.

- **xmexample1**. This is a template configuration file for a single domain.
- **xmexample2**. This is an example for multiple domain configurations in one file.

For the beginning, **xmexample1** is a better choice.

A configuration file contains several keywords which configure different aspects of a Xen domain. The following is an example configuration file using the most common options. The # character is used for comments. Please read the comments in the example for details about the configuration options.

```
# Unique name of the domain
name = "SLES10-WebServer"

# The following lines point to the kernel and initrd file
# on the filesystem of the domain. The filesystem itself is
# defined later.
kernel = "/boot/vmlinuz-Xen"
ramdisk = "/boot/initrd-Xen"

# The amount of memory that is initially assigned to the
# domain. This can be changed at runtime.
memory = 256

# The next line defines a some details about the network
# configuration. When left blank, defaults are used,
# which work fine in most cases.
vif = [ ' ' ]

# This defines the disc of the domain. "phy" means that the
# physical device /dev/hda1 is mapped to the virtual device
# /dev/hda1 in the domain. "w" indicates, that the disc is
# writable.
disk = [ 'phy:hda1,hda1,w' ]

# The following is an example for a file based filesystem
# image. In this case the "file:" keyword is used.
# disk = [ 'file:/data/vm/SLES10-disc.img,hda1,w' ]

# Sets the device for the Linux Kernel
root = "/dev/hda1 ro"
```



A good source for detailed documentation and howtos about Xen and the domain configuration files is the Xen wiki at:
<http://wiki.xensource.com/xenwiki/>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Use the xm Tool

xm is the administration tool for Xen domains. **xm** communicates with the **xend** management process running on the domain0 Linux installation.

The following is the general format of a **xm** command line:

```
xm command [options] [arguments] [variables]
```

You can get a complete list of the most common **xm** commands by entering **xm help**. A complete list can be viewed with **xm help --long**. It is also possible to display specific information about a certain command with **xm help [command_name]**.

To start a virtual machine, the **create** command is used:

```
xm create -c -f /data/xen/SLES10-WebServer.conf
```

The **-c** option lets **xm** connect to the terminal of the started domain, so that you can interact with the system. To disconnect from the terminal and return to the original command line, enter the key combination **Ctrl-]**.

The **-f** option specifies the configuration file of the domain that should be started.

The command **list** displays information about the currently running Xen domains:

```
xm list
```

The output of the **list** command contains the following fields:

- **name.** The name of the domain as specified in the configuration file.

- **domid.** A numeric, consecutive domain ID, which is automatically assigned when the domain starts.
- **memory.** The amount of memory assigned to the domain.
- **vcpus.** The number of virtual CPUs utilized by this domain.
- **state.** The current state of the domain. This could be:
 - **r.** The domain is running.
 - **b.** The domain has been created, but is currently blocked. This can happen, when a domain is waiting for I/O or when there is nothing to do for the domain.
 - **p.** The domain is paused. The state of the domain is saved and can be restored.
 - **s.** The domain is in the process of being shutdown.
 - **c.** The domain is crashed, due to an error or misconfiguration.

An alternative to list is the command **top**, which displays domain information updated in real-time.

The **console** command connects you with the terminal of a running domain:

```
xm console <domain_id>
```

The command takes the domain id as a parameter, which can be determined with the **list** command (field domid). As mentioned before, use the key combination **Ctrl-]** to disconnect from a terminal.

With the **pause** command you can interrupt the execution of a domain temporarily:

```
xm pause <domain_id>
```

A paused domain is not completely shut down. The current state is saved and the execution of the domain can be continued with the **unpause** command:

```
xm unpause <domain_id>
```

To shutdown a domain, use the **shutdown** command:

```
xm shutdown <domain_id>
```

In case the domain is not responding anymore, you can force the shutdown of the domain with the **destroy** command:

```
xm destroy <domain_id>
```

To save the state of a domain for a longer time (e.g. over a reboot of domain0) you can use the **save** command:

```
xm save <domain_id> <filename>
```

The domain can be restored from the resulting file with the **restore** command:

```
xm restore <filename>
```

Another commonly used command is **mem-set**, which allows you to change the memory allocation of a domain:

```
xm mem_set <domain_id> <amount_of_memory>
```

The amount of memory is specified in megabytes.



Instead of the domain ID <domain_id>, you can also use the domain name in all `xm` commands.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Exercise 8-3 *Change Memory Allocation of a Guest Domain*

In the following you learn how to change the memory allocation of a guest domain by changing the domain configuration file.

You can find this exercise in the workbook.

(End of Exercise)

Automate Domain Startup and Shutdown

When you start, shutdown or reboot domain0 of a Xen system, this also affects the other running Xen domains. Without a running domain0, the other Xen domains cannot operate.

SUSE Linux Enterprise Server 10 comes with a start script called **xendomains** which is included in the package **xen-tools**.

The script should be installed on domain0 and does the following:

- When domain0 is booted, all domains with configuration files located under **/etc/xen/auto/** are started.
- When domain0 is shutdown or rebooted, running Xen domains are shutdown automatically.

The script has some configuration options, which can be adjusted in the file **/etc/sysconfig/xendomains**. The configuration variables in this file are well documented.

One interesting option is to migrate domains automatically to a different host when a domain0 is shutdown. This can be configured in the variable **XENDOMAINS_MIGRATE**. The variable has to be set to the IP address of the target machine. When the variable is empty, no migration is performed.

Exercise 8-4 Automate Domain Startup

In this exercise, you learn how to startup domains automatically when the system is booted.

You can find this exercise in the workbook.

(End of Exercise)

Objective 6 Understand Xen Networking

Usually the network connection of Xen domains works out of the box. However, in case you would like to change the configuration, networking with Xen can be a bit tricky. The following should give you an overview of how Xen domains are connected to the physical network.

To better understand the concept of Xen networking, do the following:

- Understand the Basic Networking Concept
- Understand Bridging
- Understand the Network Interfaces in domain0

Understand the Basic Networking Concept

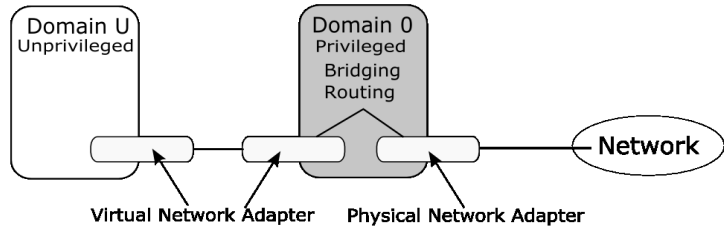
In a Xen setup, domain0 is controlling the physical network interfaces of a host system. Unprivileged domains are connected to domain0 through virtual ethernet adapters.

One virtual adapter in an unprivileged domain is connected to one virtual adapter in domain0.

In domain0, standard Linux networking mechanisms like bridging or routing are used to connect the virtual adapters through the physical adapter to the network.

The following is an illustration of this basic concept:

Figure 8-10



Understand Bridging

On SUSE Linux Enterprise Server 10, the default mechanism to connect virtual and physical interfaces in domain0 is bridging. Other mechanisms like routing with or without Network Address Translation (NAT) are out of the scope of this course.

Bridging basically means that multiple network interfaces are combined to one. Traditionally this technique is used to connect two physical network interfaces or network segments.

In a Xen system, bridging is used to connect virtual and physical network adapters in domain0. In a Xen system, you can consider the bridge as a kind of virtual switch which all virtual and physical interfaces are connected to.

The configuration of the bridge is done by the xend management process. When a new domain is created, the following changes to the network configuration are made (simplified):

1. Xen provides a virtual interface to the new domain.
2. xend creates a new virtual interface in domain0.
3. Both virtual interfaces are connected through a virtual point to point connection.

4. The virtual interface in domain0 is added to the bridge with the physical interface.

These steps only affect the general network connectivity. The IP configuration in the Xen domains has to be done separately with DHCP or a static network configuration.

xend is performing these network changes with the help of scripts, which are located at `/etc/xen/scripts/`. The following scripts are used for bridged networking:

- **network-bridge**. This script is called initially when xend is started. It sets up the bridge **xenbr0** and moves the physical interfaces onto that bridge.
- **vif-bridge**. This script is called for every domain that is started and adds the virtual interface to the bridge.

In the file `/etc/xen/xend-config.sxp` you can configure which network scripts are used by xend.

Understand the Network Interfaces in domain0

When you look at the network interfaces in domain0 with the command **ip a**, you can see that there are many more interfaces than in a regular Linux installation.

The following is an example output of **ip a** on domain0 (shortened):

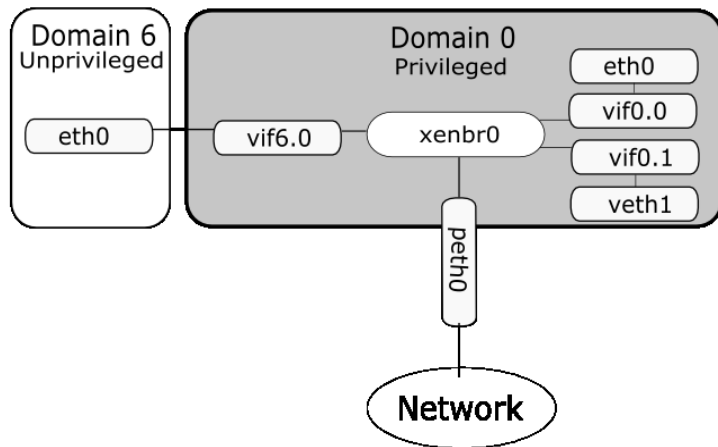
```
linux-3rsm:~ # ip a
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: peth0: <BROADCAST,MULTICAST,NOARP,UP> mtu 1500 qdisc
pfifo_fast qlen 100
    link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
    inet6 fe80::fcff:ffff:feff:ffff/64 scope link
        valid_lft forever preferred_lft forever
4: vif0.0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
    link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
    inet6 fe80::fcff:ffff:feff:ffff/64 scope link
        valid_lft forever preferred_lft forever
5: eth0: <BROADCAST,MULTICAST,NOTRAILERS,UP> mtu 1500 qdisc
noqueue
    link/ether 00:11:25:81:4c:5b brd ff:ff:ff:ff:ff:ff
    inet 149.44.171.67/23 brd 149.44.171.255 scope global
eth0
    inet6 2001:780:101:aa00:211:25ff:fe81:4c5b/64 scope
global dynamic
        valid_lft 29998sec preferred_lft 9996sec
    inet6 fe80::211:25ff:fe81:4c5b/64 scope link
        valid_lft forever preferred_lft forever
6: vif0.1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
    link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
7: veth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
[...]
19: veth7: <BROADCAST,MULTICAST> mtu 1500 qdisc noop
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
20: xenbr0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
    link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
    inet6 2001:780:101:aa00:fcff:ffff:feff:ffff/64 scope
global dynamic
        valid_lft 29998sec preferred_lft 9996sec
    inet6 fe80::200:ff:fe00:0/64 scope link
        valid_lft forever preferred_lft forever
23: vif3.0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
    link/ether fe:ff:ff:ff:ff:ff brd ff:ff:ff:ff:ff:ff
    inet6 fe80::fcff:ffff:feff:ffff/64 scope link
        valid_lft forever preferred_lft forever
```

The following interface naming schema is used in domain0:

- **peth**. These are **physical** interfaces in domain0. peth devices are connected to the network bridge.
- **vif**. These are **virtual** interfaces which are part of the bridge. The name of a vif interface identifies to which domain this interface is connected to. **For example:** vif6.0 is connected to the first virtual interface in domain 6.
- **veth**. These **virtual** interfaces are connected to the vif interfaces of domain0 (vif0.x). By default 7 vif <-> veth pairs are created. The veth interfaces can be used for more complex network setups.
- **eth0**. The first veth interface is named eth0 and connected with vif0.0. This is the “default” network interface of domain0.
- **xenbr0**. This is the default bridge that connects virtual and physical interfaces.

The following illustration gives you an overview of the interfaces in domain0.

Figure 8-11



You can use the command **brctl show** in domain0, to see which interfaces have been added to the network bridge.



Due to the complexity of the Xen network setup, the default firewall (SuSEFirewall2) is not working correctly in domain0. It is therefore recommended to disable SuSEFirewall2 and to setup a customized firewall if needed.

Exercise 8-5 Check the Network Configuration

This exercise assumes that you have a Xen system with domain 0 and one more Xen domain running.

You can find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 7 **Migrate a Guest Domain**

One advantage of virtualization is that domains can easily be moved from one physical system to another. Under Xen this procedure is called a **domain migration**.

A domain migration is performed by copying the current memory content. Please note the following before migrating a domain:

- There is no automatic way to copy the mass storage devices of a domain to another system. You have to make sure that the file systems (file system images or physical partitions) are available on the current and on the new host system. This can either be done by copying the data manually or by using a distributed file system (like NFS or SAN/NAS storage solutions).
- When a domain is migrated, the network settings are not automatically adjusted. Therefore the current and the new host system have to be in the same subnet or the network settings have to be manually adjusted after the migration.

You have the following two options to migrate a Xen domain:

- Use Domain Save and Restore
- Use Migration and Live Migration

Use Domain Save and Restore

A very simple way to migrate a domain is to use the save and restore function of the `xm` tool.

With the command **`xm save <domain_id> <filename>`**, you can suspend the specified domain and save the status to the given filename.

This file can then be copied to the new host system. To restore the domain, use the command **`xm restore <filename>`**.

As mentioned above, besides the file created with `xm`, you might also have to copy the file systems to the new host system.

Use Migration and Live Migration

Instead of the `save` and `restore` commands of `xm`, you can also use the command **`xm migrate <domain_id> <target_host>`**. This command migrates a domain directly to a new host. In this case it's not necessary to copy memory state files manually.

In order to get this working, the current and new host must be running Xen and `xend`.

By adding the option **`--live`** to the migration command line, the downtime during the migration can be reduced to typically 60-300ms. Instead of shutting down the domain before the migration starts, Xen attempts to keep it running while the migration is in progress.

The `xend` configuration file **`/etc/xen/xend-config.sxp`** contains two options concerning domain migration:

`(xend-relocation-server yes)`

This option enables the migrating functionality in `xend`.

`(xend-relocation-hosts-allow '^localhost$')`

This option controls which hosts are allowed to connect to `xend` for domain migration. By default, only `localhost` is allowed to connect. The option takes regular expressions as parameter. Have a look at the configuration file for examples.



Please note, that there are two `xend` involved in a domain migration (current and new host). You might have to adjust the `xend-config.sxp` file on both systems.

Summary

Objective	Summary
1. Understand the Concept of Virtualization	<p>Virtualization technology separates a running instance of an operating system from the physical hardware. Instead of a physical machine, the operating system runs in a so called virtual machine. Multiple virtual machines share the resources of the underlying hardware.</p> <p>Virtualization allows you to run multiple virtual systems on one single physical machine.</p>
2. Understand How Xen Works	<p>There are two different kinds of virtualization:</p> <ul style="list-style-type: none">■ Full-Virtualization■ Para-Virtualization <p>Xen uses para-virtualization. It provides access to the physical hardware through an API.</p>

Objective	Summary
3. Install Xen	<p>The following packages have to be installed in the initial SUSE Linux Enterprise Server 10 installation:</p> <ul style="list-style-type: none">■ xen. This package contains the Xen Virtual Machine Monitor (Hypervisor).■ xen-tools. Contains xend and a collection of command line tools to administer a Xen system.■ kernel-xen. This package contains a modified Linux kernel that runs in a Xen domain.■ xen-doc-* (optional). Xen documentation in various formats. <p>The installation of xen adds an entry in the grub configuration file.</p>
4. Manage Xen Domains with YaST	<p>YaST provides a module which can be used to create and manage Xen domains. The module is called: Virtual Machine Management (Xen).</p> <p>This module offers a convenient way to create and control the Xen domains on your system. The module can be started from the System section in the YaST Control Center, and has to run on the Linux system running in domain0.</p>

Objective	Summary
5. Manage Xen Domains at the Command Line	<p>Every Xen domain needs a configuration file. Usually this is located in /etc/xen/vm/.</p> <p>xm is the central administration tool for xen domains.</p> <p>To start a virtual machine, the create command is used. For example:</p> <pre>xm create -c -f SLES10.conf</pre> <p>Some services are not required in a xen environment and can be removed.</p> <ul style="list-style-type: none">■ insserv -r earlykbd■ insserv -r kbd■ insserv -r irq_balancer <p>Under Xen, all domains are connected with the physical network through domain0.</p>
6. Understand Xen Networking	<p>Domain0 is the central point to configure the network connections on a Xen system.</p> <p>A network bridge in domain0 is used as a virtual switch.</p> <p>This bridge is set up and controlled by xend.</p>

Objective	Summary
7. Migrate a Guest Domain	<p>One advantage of virtualization is that domains can easily be moved from one physical system to another. Under Xen this procedure is called a domain migration.</p> <p>Domains can be migrated with xm's save and restore commands or with the migrate command.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 9 iSCSI

The Internet Small Computer Systems Interface (iSCSI) is an implementation of the regular SCSI commands over IP. SUSE Linux Enterprise Server 10 allows to connect to an iSCSI target as an iSCSI initiator, or to offer storage space to other iSCSI initiators as an iSCSI target.

Objectives

1. iSCSI Background
2. iSCSI Configuration

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 1 iSCSI Background

SCSI commands are used to enable systems to communicate with I/O devices, especially storage devices. The Internet Small Computer Systems Interface implements the regular SCSI commands over IP.

With iSCSI, the operating system (iSCSI initiator) does not send the SCSI command directly to a device, but to another system connected via the network (iSCSI target); this system will then redirect the SCSI commands to the actual storage device—for instance a partition, a logical volume or a file.

This is basically a client/server connection between iSCSI initiator (client) and iSCSI target (server). However, unlike usual client/server connections, usually only one client may connect to the server. The iSCSI protocol is standardized in RFC3720.

With iSCSI it is possible to connect to remote storage, which can be a disk, a logical volume or a file, and it will appear to the system as a local disk. This is also known as SAN. The advantage of iSCSI in comparison to other SAN solutions is primarily in the area of cost. Common SAN solutions use rather expensive FC (fibre channel) hardware, whereas iSCSI can use the already existing ethernet network. Furthermore, many SAN solutions also provide an iSCSI interface.

Considering the amount of data transferred over the network it is important that enough bandwidth is available, for instance Gigabit Ethernet. It often makes sense to set up a dedicated network for the iSCSI connections.

The remote devices are available on Linux as a regular SCSI (/dev/sdX) device and should be mounted with the option `_netdev`. You can configure the iSCSI targets to require a username and password upon connection from an iSCSI initiator.

There is more than one iSCSI solution available on Linux, but all of them use by default port 3260 for the iSCSI connection.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 2 iSCSI Configuration

The configuration and maintenance of iSCSI on RHEL4 and SUSE Linux Enterprise Server 10 differ in some aspects.

- Red Hat Enterprise Linux 4
- SUSE Linux Enterprise Server 10

Red Hat Enterprise Linux 4

With RHEL, iSCSI is available since Service Pack 2. It is implemented by two modules, `iscsi_sfnet` and `scsi_transport_iscsi`, and the package `iscsi-initiator-utils`.

To see if any iSCSI devices are currently used, use the command **`iscsi-ls -l`**.

If there are any devices listed, then you should analyze and save the configuration files used for the iSCSI connection: **`/etc/ietd.conf`**, **`/etc/iscsi.conf`** and **`/etc/initiatorname.iscsi`**. Their role will be described in detail later in this section.

Before you migrate to SUSE Linux Enterprise Server 10, you should stop the iSCSI daemon with the command **`service iscsi stop`**.

You should also check `/etc/fstab` for any configured iSCSI devices.

SUSE Linux Enterprise Server 10

SUSE Linux Enterprise Server 10 can be configured as iSCSI target and as iSCSI initiator. You can do this with the configuration tool YaST or manually.

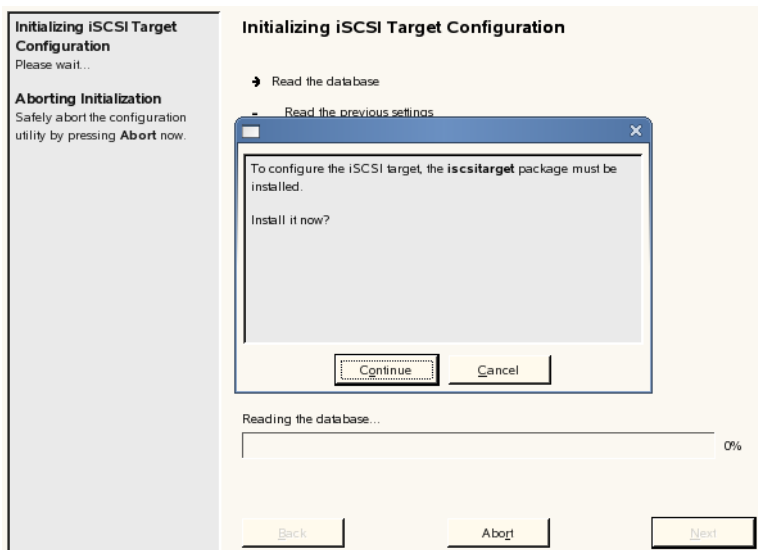
- Set up an iSCSI Target
- Set up an iSCSI Initiator
- Mount iSCSI Targets Automatically at Boot Time

Set up an iSCSI Target

You can either start YaST and select **Network Services > iSCSI Target**, or start the iSCSI Target module directly by entering as root in a console window **yast2 iscsi-server**.

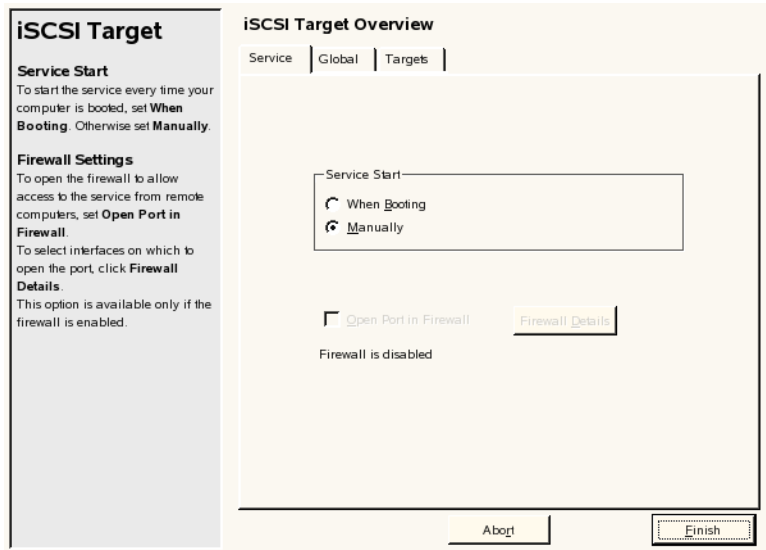
If the `iscsitarget` package is not installed, YaST will invite you to do that:

Figure 9-1



In the main menu of the iSCSI Target configuration there are 3 tabs available. Service is the first one; here you can configure how iSCSI will be started. Most likely **When Booting** is the right choice for you.

Figure 9-2



If your SuSE Firewall is active, you should select **Open Port in Firewall** to be able to access the port needed for iSCSI. If the Firewall is not activated, the option cannot be selected, as visible in the above screenshot.

On the **Global** register you can configure the authentication between iSCSI target and initiator during the discovery process.

Figure 9-3

The screenshot shows the 'iSCSI Target Overview' window with the 'Global' tab selected. On the left, a sidebar titled 'iSCSI Target' provides instructions: 'Select the type of authentication. Use **No Authentication** or one of **Incoming** and **Outgoing** (can be both together). Then insert **User** and **Password**. For incoming authentication, it is possible to **Add** more pairs and **Edit** and **Delete** them.' The main panel has three tabs: 'Service', 'Global', and 'Targets'. Under 'Global', there are three sections: 'No Authentication' (checked), 'Incoming Authentication' (unchecked), and 'Outgoing Authentication' (unchecked). The 'Incoming Authentication' section has 'Username' and 'Password' input fields and 'Add', 'Edit', and 'Delete' buttons. The 'Outgoing Authentication' section also has 'Username' and 'Password' input fields. At the bottom right are 'Abort' and 'Finish' buttons.

By default **No Authentication** is selected. That means that every iSCSI client can discover this target.

The difference between **Incoming** and **Outgoing Authentication** is as follows:

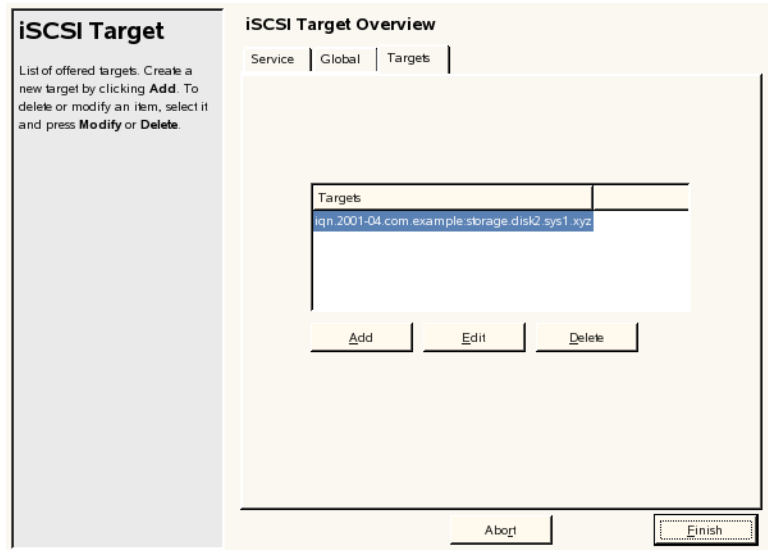
Incoming Authentication signifies that the iSCSI initiator is responsible for the authentication. If you use **Outgoing Authentication**, then the iSCSI target is responsible. Challenge-Handshake Authentication Protocol (CHAP) is used for authentication.

The passwords defined here will be stored in the configuration files as clear text.

You can also set up the iSCSI environment without authentication (default).

On the third register you can configure your iSCSI targets:

Figure 9-4



By default there is already an iSCSI target defined, but because there is no backend configured, it is not directly usable. You could edit this target, but you can also delete it and create a new one.

Before you add an iSCSI target you should decide on the backend device you want to use. As backend device you can use a regular disk, a logical volume or a file. To demonstrate the configuration we will use a file with a size of 1 GB, which can be created with the following commands:

```
mkdir /srv/iscsi
```

```
dd if=/dev/zero of=/srv/iscsi/first-device bs=1M count=1024
```

Selecting the **Add** button opens the following dialog:

Figure 9-5

iSCSI Target

Create a new target. Replace template values with the correct values. For **Target**, use the format iqn.yyyy-mm.. For **Path**, use block devices, regular files, LVM, or RAID.

Add iSCSI Target

Target iqn.2006-10.com.example	Identifier 9d60efb1-7a9e-481a-95b5-68da4ef1b6e9
LUN 0	Path /tmp/file

Abort Next

In this dialog you can define the name of the iSCSI device and the backend to use. To use more LUNs (Logical Unit Numbers) on one iSCSI target you have to edit the file `/etc/ietd.conf` manually.

The target name should be unique world wide. By default, the following syntax is used:

iqn.YYYY-MM.reverse-domain-name: identifier

iqn (iSCSI qualified name) is normally followed by the date of device creation and the domain name of the company in reverse order. The identifier is optional and can be used for a name based identification, like for example “iscsi-database-target”.

In the **path** box you define the backend device; in our case this would be `/srv/iscsi/first-device`.

Selecting **Next** opens the next dialog where you can configure the authentication for this target.

Figure 9-6

Select the type of authentication. Use **No Authentication** or one of **Incoming** and **Outgoing** (can be both together). Then insert **User** and **Password**. For incoming authentication, it is possible to **Add** more pairs and **Edit** and **Delete** them.

Modify iSCSI Target

☒ No Authentication

☐ Incoming Authentication

Username	Password

Add

Edit

Delete

☐ Outgoing Authentication

Username

Password

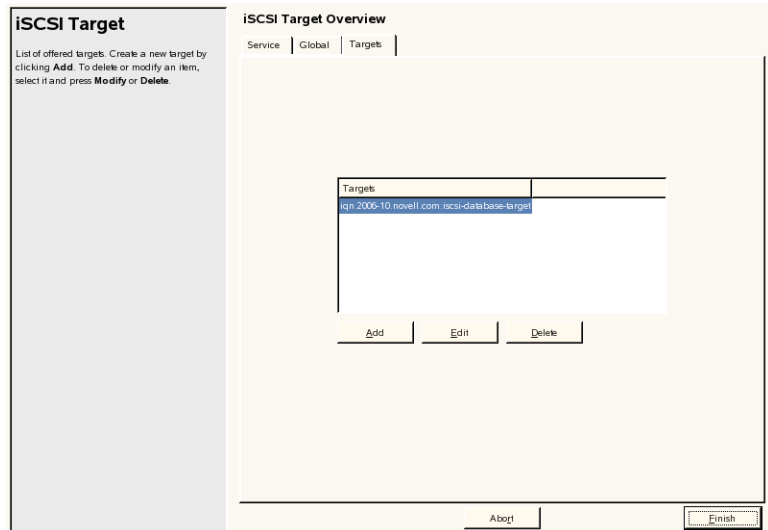
Abort

Next

By default there is no authentication used. If you add a user and password for **Incoming Authentication**, you have to remember user and password for the iSCSI initiator setup.

After selecting **Next** you can view your new iSCSI target:

Figure 9-7



After selecting **Finish** the settings are written to the system.

The command **rciscsitarget status** (or **/etc/init.d/iscsitarget status**) displays whether the iSCSI target daemon is running or not. If it is, **cat /proc/net/iet/volume** will display your configured iSCSI target.

The iSCSI Target configuration is stored in **/etc/ietd.conf** and is managed by the service **iscsitarget-daemon** (**ietd**: iSCSI Enterprise Target Daemon).

The following listing shows an `/etc/ietd.conf` with two targets; one uses the device `/dev/sda6` as physical storage, the other one a file. The comments explain further parameters you might want to define for each target:

```
Target iqn.2006-10.com.digitalairlines:878137d1-d58f ...
Lun 0 Path=/dev/sda6,Type=fileio
Target iqn.2006-10.com.digitalairlines:14736b2d-f00e ...
Lun 1 Path=/srv/iscsi/second-device,Type=fileio
    # Users, who can access this target. The same rules
    # as for discovery users apply here. Leave them
    # alone if you don't want to use authentication.
    #IncomingUser joe secret
    #OutgoingUser jim 12charpasswd
    # Logical Unit definition
    # You must define one logical unit at least.
    # Block devices, regular files, LVM, and RAID can
    # be offered to the initiators as a block device.
    #Lun 0 Path=/dev/sdc,Type=fileio
    # Alias name for this target
    # Alias Test
    # various iSCSI parameters
    # (not all are used right now, see also iSCSI spec
    # for details)
    #MaxConnections          1
    #InitialR2T              Yes
    #ImmediateData          No
    #MaxRecvDataSegmentLength 8192
    #MaxXmitDataSegmentLength 8192
    #MaxBurstLength          262144
    #FirstBurstLength        65536
    #DefaultTime2Wait        2
    #DefaultTime2Retain      20
    #MaxOutstandingR2T       8
    #DataPDUInOrder          Yes
    #DataSequenceInOrder     Yes
    #ErrorRecoveryLevel      0
    #HeaderDigest             CRC32C,None
    #DataDigest               CRC32C,None
    # various target parameters
    #Wthreads                 8
```

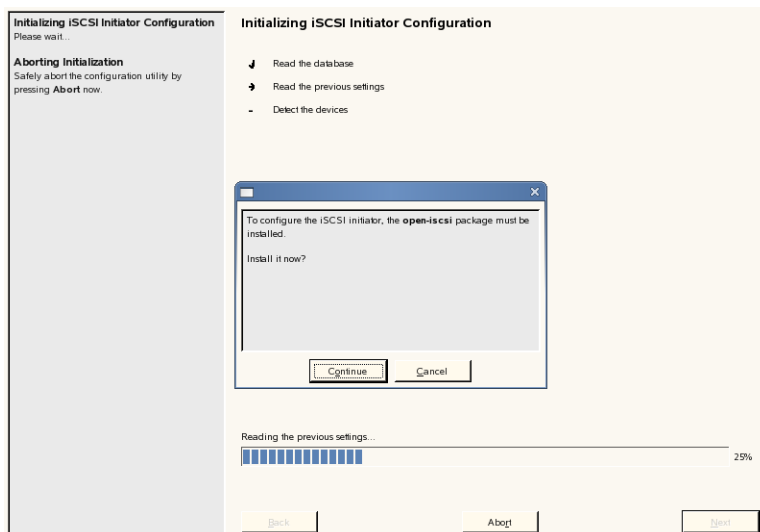
1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Set up an iSCSI Initiator

You can either start YaST and select **Network Services > iSCSI Initiator**, or start the iSCSI Initiator module directly by entering as root in a console window **yast2 iscsi-client**.

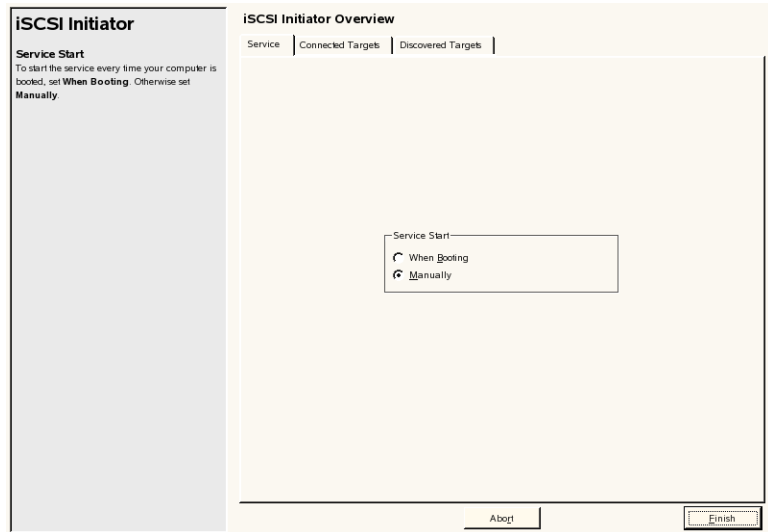
If the open-iscsi package is not installed, YaST will invite you to do that.

Figure 9-8



After the installation, the following dialog opens up:

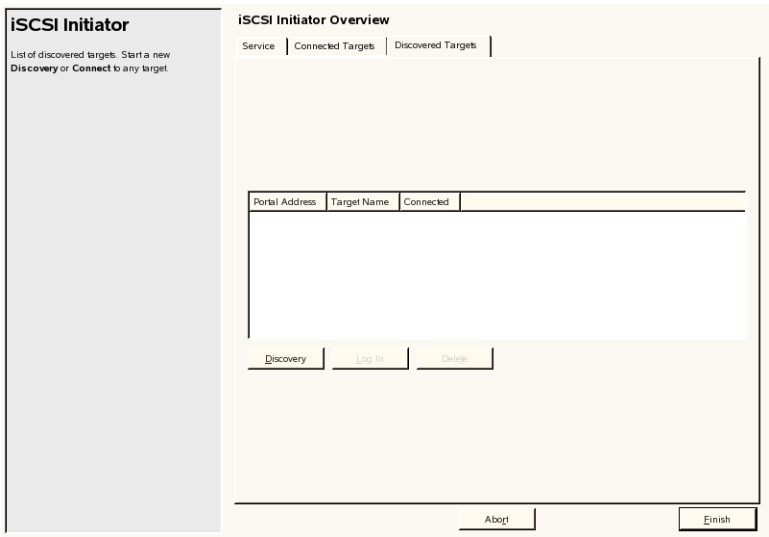
Figure 9-9



In the **Service** tab you can configure whether the iSCSI client should start automatically at boot time or manually.

The **Connected Targets** tab shows the active connections. But first you have to discover and configure an iSCSI target device. To do this, select the **Discovered Targets** tab:

Figure 9-10



Currently there are no target devices discovered. With the **Discovery** button you can start a search on a specific IP address.

Figure 9-11

iSCSI Initiator

Enter the **IP Address** of the discovered server. Only change **Port** if needed. For authentication, use **Username** and **Password**. If you do not need authentication, select **No Authentication**.

Warning

When accessing an iSCSI device **READ/WRITE**, make sure that this access is exclusive. Otherwise there is a potential risk of data corruption.

iSCSI Initiator Discovery

IP Address Port

☒ No Authentication

☐ Incoming Authentication

Username Password

☐ Outgoing Authentication

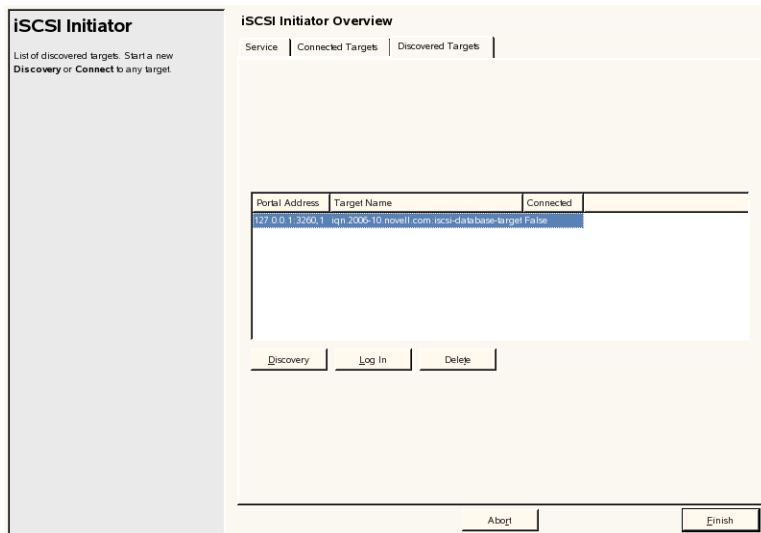
Username Password

Abort Next

Enter the IP address you want to scan for iSCSI target devices. If username and password are needed for the discovery, enter them here. By default you don't have to use any authentication.

You can scan your own system by entering the IP address 127.0.0.1. If an iSCSI target device is available on the tested system, a dialog similar to the following opens up:

Figure 9-12

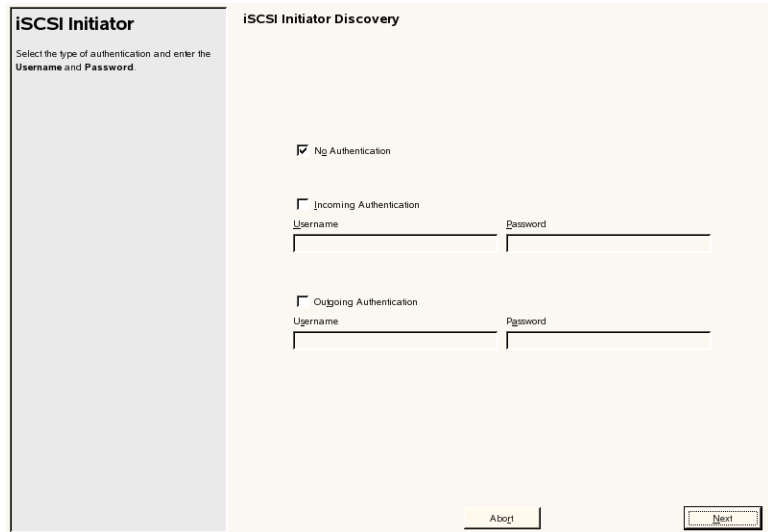


If there are several iSCSI target devices available, they will all get listed. The iqn (iSCSI Qualified Name) is used to identify the target.

After selecting the iSCSI target from the list you have to log in using the **Log In** button.

The following dialog appears:

Figure 9-13

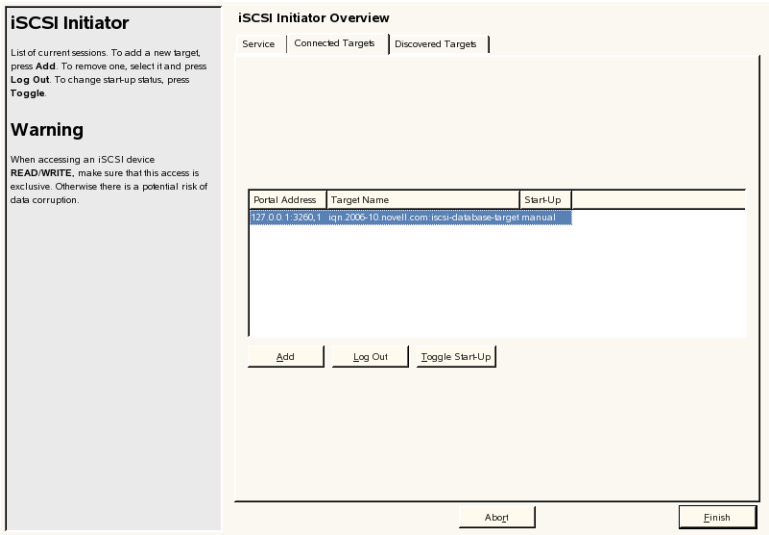


The dialog box is titled "iSCSI Initiator" on the left and "iSCSI Initiator Discovery" on the right. The left pane contains the text "Select the type of authentication and enter the Username and Password." The right pane has a checked checkbox for "No Authentication". Below it, there are two unchecked checkboxes: "Incoming Authentication" and "Outgoing Authentication". Each of these has associated "Username" and "Password" text input fields. At the bottom right, there are "Abort" and "Next" buttons.

By default there is no authentication necessary. However, if you have, for example, configured your iSCSI target device with incoming authentication, then you have to select **Outgoing Authentication** in this dialog and enter the same username and password which you used for the iSCSI target configuration.

After selecting **Next**, the **Connected Targets** tab displays your configured iSCSI devices.

Figure 9-14



With the **Toggle Start-Up** button you can switch between **automatic** and **manual**, which defines how the device is activated.

Select **Finish** to write the settings to the system.

The client configuration is stored in the file `/etc/iscsi.conf` and within the directory `/var/lib/open-iscsi/` in the database files `discovery.db` and `node.db`. The tool to use for manual configuration of these files is **iscsiadm**.

The name of the iSCSI initiator is contained in the file
`/etc/initiatorname.iscsi`:

```
da3:~ # cat /etc/initiatorname.iscsi
## DO NOT EDIT OR REMOVE THIS FILE!
## If you remove this file, the iSCSI daemon will not start.
## If you change the InitiatorName, existing access control lists
## may reject this initiator. The InitiatorName must be unique
## for each iSCSI initiator. Do NOT duplicate iSCSI InitiatorNames.
InitiatorName=iqn.1996-04.de.suse:01.427f8cb385f5
```

A description of how `iscsid`, `iscsiadm` and the configuration files
play together is contained in
`/usr/share/doc/packages/open-iscsi/README`.

In the file `/var/log/messages` you should see entries similar to the
following:

```
Oct  4 11:55:56 da3 kernel: scsi3 : iSCSI Initiator over TCP/IP, v1.0-595
Oct  4 11:55:56 da3 kernel: sess_param(173) 1 1 1 8192 8192 262144 65536 2
0 1 1 1 0 1 1
Oct  4 11:55:56 da3 kernel:   Vendor: IET           Model: VIRTUAL-DISK
Rev: 0
Oct  4 11:55:56 da3 kernel:   Type:   Direct-Access
ANSI SCSI revision: 04
Oct  4 11:55:56 da3 kernel: SCSI device sdb: 2097152 512-byte hdwr sectors
(1074 MB)
Oct  4 11:55:56 da3 kernel: sdb: Write Protect is off
Oct  4 11:55:56 da3 kernel: sdb: Mode Sense: 77 00 00 08
Oct  4 11:55:56 da3 kernel: SCSI device sdb: drive cache: write through
Oct  4 11:55:56 da3 kernel: SCSI device sdb: 2097152 512-byte hdwr sectors
(1074 MB)
Oct  4 11:55:56 da3 kernel: sdb: Write Protect is off
Oct  4 11:55:56 da3 kernel: sdb: Mode Sense: 77 00 00 08
Oct  4 11:55:56 da3 kernel: SCSI device sdb: drive cache: write through
Oct  4 11:55:56 da3 kernel:   sdb: unknown partition table
Oct  4 11:55:56 da3 kernel: sd 3:0:0:0: Attached scsi disk sdb
Oct  4 11:55:56 da3 kernel: sd 3:0:0:0: Attached scsi generic sg2 type 0
Oct  4 11:55:56 da3 iscsid: version 1.0-604
Oct  4 11:55:56 da3 iscsid: iSCSI daemon with pid=6172 started!
Oct  4 11:55:56 da3 iscsid: iSCSI sync pid=6173 started
Oct  4 11:55:56 da3 iscsid: connection0:0 is operational now
```

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

You can see that there is a new SCSI device available (/dev/sdb). You can create partitions using `fdisk`, create file systems on the partitions and mount them.

With the command `ls SCSI` you can see all available SCSI devices.

In case you want to see whether any active iSCSI connection exists, have a look into the file `/proc/net/iet/session` on the server hosting the target using `cat`. You can see all iSCSI connections to a target on this host, as in the following example:

```
da3:~ # cat /proc/net/iet/session
tid:1
name:ign.2006-10.com.digitalairlines:d3d43303-8b2c-43df-83ea-6721942677e2
      sid:281474980708864 initiator:ign.1996-04.de.suse:01.427f8cb385f5
      cid:0 ip:127.0.0.1 state:active hd:none dd:none
```

You can see that a connection from the IP address 127.0.0.1 is established to the Target with the id 1 (tid).

While it is possible to have more than one connection you should keep in mind that it is not recommended to mount an iSCSI target more than once without any precautions (like a cluster file system on top of the iSCSI target).

Mount iSCSI Targets Automatically at Boot Time

On the server (iSCSI target) side you have to make sure that the `iscsi` target service will be started automatically after a reboot. Use the command `chkconfig iscsitarget on` (or `insserv iscsitarget`) to do this.

On the client (iSCSI initiator) side the `open-iscsi` daemon has to start automatically at boot time. This is achieved using the command `chkconfig open-iscsi on`.

If you checked the button **When booting** in the respective YaST dialog, it is not necessary to issue the above `chkconfig` commands.

Generally you enter all the devices which should be mounted automatically at boot time into the file `/etc/fstab`. One problem you face with iSCSI devices is that no network is available when the file `/etc/fstab` is read during the boot process. Another problem is that you do not know for sure which local SCSI device will be used for the iSCSI target.

There is a solution for both problems. To be sure that the iSCSI device will be mounted to the same directory every time you have to use the UUID (Unique Universal ID) from the partition. To establish the UUID of a device use the following command:

```
da3:~ # udevinfo -q symlink -n /dev/sdb1
disk/by-id/scsi-1494554000000000000000000010000005d9a01000
2008000-part1
disk/by-path/ip-127.0.0.1:3260-iscsi-ign.2006-10.com.digit
alairlines:d3d43303-8b2c-43df-83ea-6721942677e2-part1
disk/by-uuid/8b4dcb80-372d-44ba-a35b-96b5619b8237
```

In this example, the last line contains the UUID of the device `/dev/sdb1`. Now you can create an entry like the following in `/etc/fstab` to mount this iSCSI device to the `/iscsi` directory:

```
/dev/disk/by-uuid/8b4dcb80-372d-44ba-a35b-96b5619b8237
/iscsi auto hotplug,defaults 0 0
```

With the mount option **hotplug** the first problem is also solved. This option signifies that the hotplug daemon, which is started after the network, will take care of this mount.

Exercise 9-1 *Set up an iSCSI Target and an iSCSI initiator*

In this exercise, you learn how to set up an iSCSI target and how to access that target using an iSCSI initiator.

You will find this exercise in the workbook.

(End of Exercise)

Summary

Objective	Summary
1. iSCSI Background	<p>The Internet Small Computer Systems Interface implements the regular SCSI commands over IP.</p> <p>The server offering storage is called iSCSI target, the client using that storage is called iSCSI initiator.</p>
2. iSCSI Configuration	<p>iSCSI target and initiator can easily be configured using the respective YaST modules.</p> <p>The iSCSI Target configuration is stored in <code>/etc/ietd.conf</code>.</p> <p>The iSCSI initiator configuration is stored in <code>/etc/iscsi.conf</code>, <code>/etc/initiatorname.iscsi</code>, and in the files <code>discovery.db</code> and <code>node.db</code> in <code>/var/lib/open-iscsi</code>.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION 10 Cluster File Systems

With a cluster file system it is possible to share a file system between systems. In the cluster jargon these systems are called nodes. Cluster file systems are also known as “shared disk file systems”. The benefit of these file systems is that data is still available to the remaining nodes if one node fails. Cluster file systems are routinely used in high availability environments.

Different cluster file systems are available for Linux, the best known are:

- OCFS (Oracle Cluster File System)
- GPFS (General Parallel File System)
- GFS (Global File System)
- LUSTRE

Objectives

1. Global File System (GFS)
2. Oracle Cluster File System 2 (OCFS2)

Objective 1 **Global File System (GFS)**

Under RHEL, primarily GFS is used as cluster file system. GFS is a cluster file system that was originally developed at the University of Minnesota, later distributed by Sistina Software. Red Hat acquired Sistina Software at the end of 2003 and released their cluster file system as Open Source Software under the GPL in 2004.

The Red Hat Cluster Suite for RHEL4 utilizes GFS 6.1 for shared storage for its high-throughput and high-availability features, but it can be used and is supported for the Oracle Real Application Cluster (RAC) as well.

All releases up to the current iteration 6.1 are on-disk compatible and run under the GFS label. The successor GFS2, currently in development, will not be on-disk compatible with GFS(1), but will eliminate some of its current caveats. GFS2 has recently been included into the development version of the Linux kernel 2.6.19; GFS1 is developed out-of-tree.

One can use GFS as a simple local file system without any cluster locking mechanism, but usually it is run in cluster mode, where it has to be supplemented by a SAN environment.

GFS is not available on SUSE Linux Enterprise Server 10.

Objective 2 Oracle Cluster File System 2 (OCFS2)

This objective covers

- OCFS2 Background
- OCFS2 Configuration
- OCFS to OCFS2 Migration
- GFS - OCFS2 Comparison Table

OCFS2 Background

OCFS2 is a cluster file system that was released by Oracle as Open Source software under the GPL in 2005. Contrary to its predecessor OCFS, released in 2002, it is now a general-purpose cluster file system that is suitable as a storage base for a more widespread type of scenarios than deploying Oracle Real Application Cluster (RAC) only.

In SUSE Linux Enterprise Server 10 it can, for instance, be used to give several hosts access to XenVM loopback disk images. Thanks to OCFS2 virtual machines can be relocated from host to host.

Novell has integrated OCFS2 version 1.0.8 into SLES 9 SP2, version 1.2.1 into SP3 and SLES 10. As OCFS2 has been included into the mainline Linux kernel starting with version 2.6.16, probably more Linux distributions will integrate OCFS2 in the future.

All OCFS2 versions—starting with version 1.0—are on-disk compatible, but their corresponding tools are not. So take care to upgrade them together, and access the cluster filesystem only with management tools that are at the same level as the kernel driver.

On SLES10 it is possible to set up an OCFS2 from the command line or with the shipped OCFS console which provides a GUI.

OCFS2 has the following features:

- POSIX compliant
- Support for multiple CPU architectures as it is endian-safe.
- Thread-safe, asynchronous I/O, metadata and data caching provides high performance.
- Lockless node-local allocation area enables fast creation of new files.
- Extents-based; allocates metadata groups automatically, no fixed inode limit.
- Direct I/O to bypass local OS cache.
- Fully integrated with the Heartbeat 2 cluster solution in SUSE Linux Enterprise Server 10.
- OCFS2 is certified for Oracle 9iR2, 10g, 10gR2 RAC on RHEL4, SLES 9 SP3 and SLES 10.

OCFS2 Configuration

The following sample configuration assumes that the OCFS2 resides on the device “sda”. To be able to share this device between nodes, this device could be made available using iSCSI as covered in Section “iSCSI” on page 9-1.



To avoid having the storage device as a possible single point of failure, which would thwart the purpose of a HA cluster, the storage device should include HA features as well. DRBD (Distributed Replicated Block Device) and Heartbeat could be used for this purpose; configuration of DRBD and Heartbeat is, however, beyond the scope of this course.

Most of the following steps to configure OCFS2 need to be taken on every node:

1. Create SSH keys for graphical OCFS2 frontend `ocfs2console`.

To facilitate the configuration on the different nodes, one usually uses ssh keys.

Keys are generated by the command **ssh-keygen**. The option **-N ""** creates a key without a passphrase. **ssh-copy-id** is a script that adds the public key to the `.ssh/authorized_keys` file on the destination computer.

```
da10:~ # ssh-keygen -t rsa -N "";
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): <ENTER>
Created directory '/root/.ssh'.
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
cb:1e:a9:7b:08:a7:82:d5:24:fc:90:49:cc:84:d6:a7 root@da10

da10:~ # ssh-copy-id -i ~/.ssh/id_rsa.pub root@da11;
da10:~ # ssh-copy-id -i ~/.ssh/id_rsa.pub root@da12;
...
```

2. Install the OCFS2 packages.

Use the YaST Software Management module or the command **yast -i ocfs2-tools ocfs2console**.

3. Create /etc/ocfs2/cluster.conf

The following is an example with two nodes, just adapt "node_count" if you add more.

```
cluster:
    name = mycluster
    node_count = 2
node:
    name = da10
    cluster = mycluster
    number = 0
    ip_address = 10.0.0.10
    ip_port = 7777
node:
    name = da11
    cluster = mycluster
    number = 1
    ip_address = 10.0.0.11
    ip_port = 7777
```

Instead of the hostname, **name** could also be the fully qualified domain name; name resolution has to work.

4. Create an OCFS2 on the shared block device.

This is only done on one node.

Take care that you use the same label as in the cluster configuration file (this is **mycluster** in the example above).

The command to create an OCFS2 file system is **mkfs.ocfs2**. Entering the command without options gives a brief syntax overview:

```
da3:~ # mkfs.ocfs2
Usage: mkfs.ocfs2 [-b block-size] [-C cluster-size]
[-N number-of-node-slots] [-T filesystem-type]
[-L volume-label] [-J journal-options] [-HFqvV] device [blocks-count]
```

The manual page explains the available options.

The following command creates an OCFS2 file system on /dev/sda:

```
da10:~ # mkfs.ocfs2 -b 4k -C 128k -N 4 -L mycluster /dev/sda
mkfs.ocfs2 1.2.1
Filesystem label=mycluster
Block size=4096 (bits=12)
Cluster size=131072 (bits=17)
Volume size=8595308544 (65577 clusters) (2098464 blocks)
3 cluster groups (tail covers 1065 clusters, rest cover 32256 clusters)
Journal size=67108864
Initial number of node slots: 4
Creating bitmaps: done
Initializing superblock: done
Writing system files: done
Writing superblock: done
Formatting Journals: done
Writing lost+found: done
mkfs.ocfs2 successful
```

5. Add an entry in /etc/fstab.

The following is done on all nodes and assumes that the device (/dev/sda in this example) is the same on all nodes

```
da10:~ # echo '/dev/sda /mnt ocfs2 defaults 0 0' >> /etc/fstab
```

6. Configure the OCFS2 cluster services.

Make sure that **Cluster to start on boot** is the same as the entry in `/etc/ocfs2/cluster.conf`

```
da10:~ # /etc/init.d/o2cb configure
Configuring the O2CB driver.
This will configure the on-boot properties of the O2CB driver.
...

Load O2CB driver on boot (y/n) [y]: <ENTER>
Cluster to start on boot (Enter "none" to clear) [mycluster]: <ENTER>
Use user-space driven heartbeat? (y/n) [n]: <ENTER>
Writing O2CB configuration: OK
```

7. Start the storage cluster services automatically.

The following commands create the links in `/etc/init.d/rcx.d/` needed to start the services at boot time.

```
da10:~ # chkconfig -a o2cb
da10:~ # chkconfig -a ocfs2
```

8. Start the OCFS2 cluster services manually and mount the OCFS2.

The following commands start the services needed for ocfs2:

```
da10:~ # /etc/init.d/o2cb start;
Loading module "configfs": OK
Mounting configfs filesystem at /sys/kernel/config: OK
Loading module "ocfs2_nodemanager": OK
Loading module "ocfs2_dlm": OK
Loading module "ocfs2_dlmfs": OK
Mounting ocfs2_dlmfs filesystem at /dlm: OK
Starting cluster mycluster: OK

da10:~ # /etc/init.d/ocfs2 start
Starting Oracle Cluster File System (OCFS2)      done
```

The second command mounts the file system. Use the command **mount** to see the result.

9. Check that the cluster file system works.

```
da10:~ # cat
/sys/kernel/config/cluster/mycluster/heartbeat/dead_threshold
7
```

10. Stop the OCFS2 cluster services manually and unmount the OCFS2.

The following commands stop the OCFS2 cluster services:

```
da10:~ # /etc/init.d/ocfs2 stop;
Stopping Oracle Cluster File System (OCFS2)      done

da10:~ # /etc/init.d/o2cb stop;
Cleaning heartbeat on mycluster: OK
Stopping cluster mycluster: OK
Unloading module "ocfs2": OK
Unmounting ocfs2_dlmfs filesystem: OK
Unloading module "ocfs2_dlmfs": OK
Unmounting configfs filesystem: OK
Unloading module "configfs": OK
```

You should also familiarize yourself with the graphical OCFS2 frontend **ocfs2console** which allows to adminster different aspects of OCFS2.

Figure 10-1

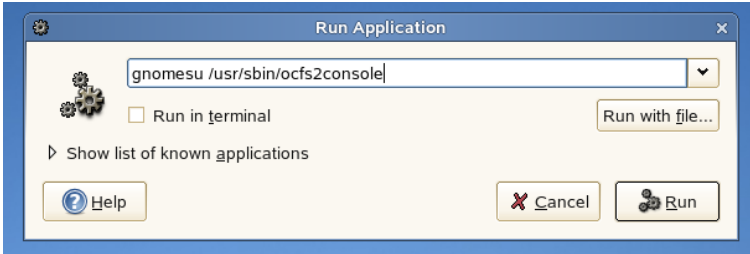
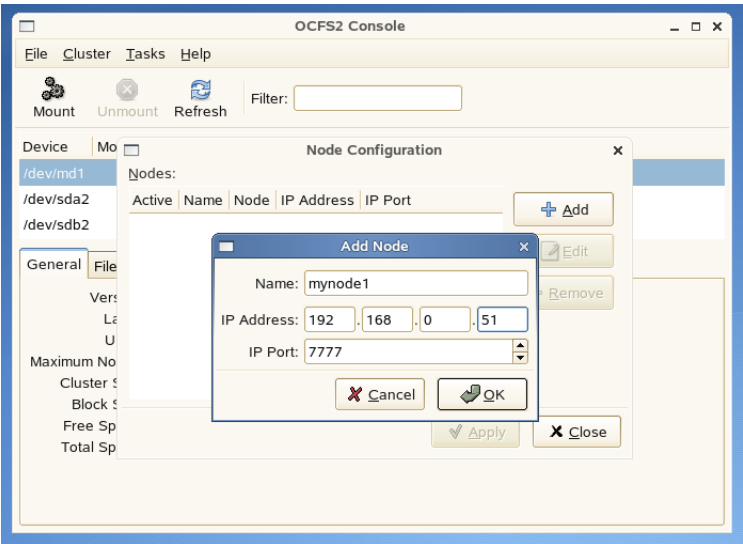


Figure 10-2



OCFS to OCFS2 Migration

The on-disk layouts of OCFS and OCFS2 are not compatible. As drivers for OCFS only exist for 2.4.x linux kernels and drivers for OCFS2 only for 2.6.x linux kernels, you cannot use them both on the same machine concurrently.

There are two possibilities to migrate to the new OCFS2:

- If the old cluster file system is still online you can replicate it to the new one via the network. You have to ensure that there are no changes to the old OCFS storage cluster after the last replication; during the migration you need double SAN capacity.
- Oracle provides the FSCat tools that can read OCFS disk images in userland. Using these tools on an OCFS2 cluster node, you can extract your OCFS backup into the OCFS2 mountpoint.

GFS - OCFS2 Comparison Table**Table 10-1**

	GFSv1 6.1	OCFS2 1.2.3
Release date	2006-08 (v1.0 1996)	2006-07 (v1.0 2005)
Linux kernel inclusion	out-of-tree	2.6.16
CPU architecture	x86, x64, ia64	x86, x64, ia64, ppc, ppc64
Max. supported nodes	256	255
Max. file name length	255	255
Max. path length	unlimited	unlimited
Block size	512 B - 4 KB	512 B - 4 KB
Cluster size	-	4 KB - 1 MB
Max. file size	2 TB / 16 TB / 8 EB	4 TB / 16 TB / 4 PB
Max. fs size	2 TB / 16 TB / 8 EB	4 TB / 16 TB / 4 PB
Volume manager	LVM2 + CLVM	LVM2, EVMS
Lock manager	GULM, kernel DLM	OCFSv2 DLM
Cluster manager	CMAN	O2CB (NM+HB+TCP)
Shared block devices	GNBD, FC, iSCSI	FC, iSCSI
Online/Offline resizing	grow/-	-/-
User/Group quotas	yes/yes	-/-
Multipath	yes	yes
Direct I/O	yes	yes

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Exercise 10-1 Set up an OCFS2

The purpose of this exercise is to familiarize you with OCFS2.

This exercise builds on the previous one, “Set up an iSCSI Target and an iSCSI initiator” on page 9-23.

You will find this exercise in the workbook.

(End of Exercise)

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Summary

Objective	Summary
1. Global File System (GFS)	Under RHEL, primarily GFS (Global File System) is used as cluster file system.
2. Oracle Cluster File System 2 (OCFS2)	<p>OCFS2 is the cluster file system used under SUSE Linux Enterprise Server 10.</p> <p>The configuration is contained in <code>/etc/ocfs2/cluster.conf</code>; the program to create the file system is mkfs.ocfs2.</p> <p>The clustering functionality is provided by the services o2cb and ocfs2.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION A Appendix: AutoYaST

This section explains how to perform an automated installation using the AutoYaST feature of SUSE Linux Enterprise Server 10.

AutoYaST facilitates the roll-out of large numbers of machines. The installation is controlled by an XML file which contains the machine specific information, e.g. IP address, hostname, partitioning, etc. Manual intervention during the installation process is unnecessary.

The installation is started using a CD, a boot floppy disk or a PXE capable network card.

The installation source can be the set of CDs as well as an installation server in the network. The supported protocols for accessing the repository on the installation server are NFS, HTTP and FTP.

Objectives

1. Autoinstallation Basics
2. Set up an Installation Server
3. Create a Configuration File for AutoYaST
4. Start the Installation

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective 1 **Autoinstallation Basics**

Both Red Hat and SUSE Linux Enterprise Server 10 provide a way to automate installations:

- Kickstart on RHEL4
- AutoYaST on SUSE Linux Enterprise Server 10

Kickstart on RHEL4

Red Hat uses the anaconda tool for the installation. If you set up the installation automatically this is referred to as kickstart environment.

The kickstart configuration file is a regular text file that contains the answers to installation questions in separate lines.

When installing a Red Hat system, a kickstart configuration file `anaconda-ks.cfg` will be created in the home directory of the root user (`/root`). This file contains the settings you entered during the installation. Before using it you have to uncomment the lines containing the partition setup.

To create a kickstart file from scratch you would use the tool `system-config-kickstart`. Manual editing of the file is required if you want setup software raid or LVM.

AutoYaST on SUSE Linux Enterprise Server 10

AutoYaST is the tool for automated installations on SUSE Linux Enterprise Server 10. All information needed during installation, e.g. partitioning or software selection, is provided by a control file in XML format. No manual intervention is necessary during the installation process.

If you have to install several systems with the same setup you can save time by automating the installation. Depending on your requirements, you can ensure all systems are set up with the same configuration, or configure systems individually with specific control files.

You should not confuse auto installation with cloning or imaging. An automated installation is a regular installation where answers to questions asked during the installation are contained in the control file. The hardware detection is still done so that the same control file can be used on diverse hardware. Imaging or cloning require identical hardware of source and target of the image.

AutoYaST is optimally used in conjunction with an installation server also providing a TFTP and a DHCP server. The advantages are:

- To start the installation, you only have to insert a suitable boot disk.
- The computer receives all information necessary for the installation via the network.
- In conjunction with wake-on-lan and/or PXE boot-enabled network interface cards, not even a boot disk is required.
- Even the on-site attendance of an administrator is not necessary for the installation.

The installation server can be accessed via the protocols NFS, HTTP and FTP.

This results in a highly simplified installation of a large number of individually configured computers.

AutoYaST can also be used to copy additional files into the installed system, as well as scripts which are executed at the end of the installation.

It is also possible to create a control file at installation time. In the last menu of the installation process you can check the box **Clone This System**. This will create an `autoinst.xml` file in the home directory of the root user (`/root`). The creation of an AutoYaST control file using the YaST AutoYaST module is covered later in this section.



In this section, we will not cover a fully automated installation. Instead, we will use the CD 1 as a boot medium and access the installation repository via NFS. The control file will be provided on a floppy disk.

Objective 2 Set up an Installation Server

AutoYaST requires a certain layout of directories and files on the installation server.



The layout is described in http://www.suse.com/~ug/AutoYaST_FAQ.html.

The purpose of the YaST module “Installation Server” is to create an installation repository that is suitable for automated installation using AutoYaST.

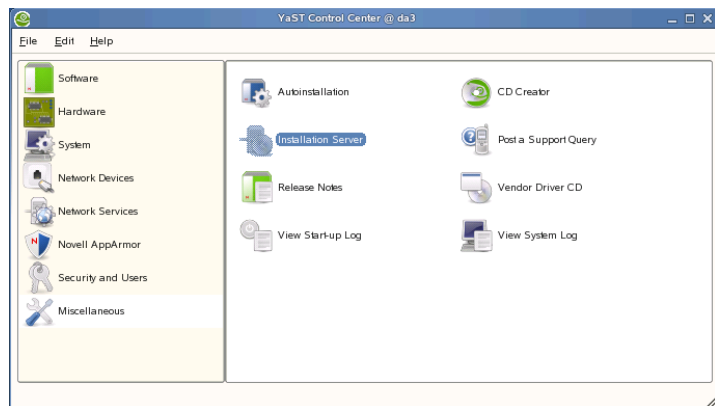
It copies the needed files to their proper location and creates certain links and files.

For setting up the installation server, start the YaST module “Installation Server”.

1. Select

yast2 > Miscellaneous > Installation Server

Figure A-1



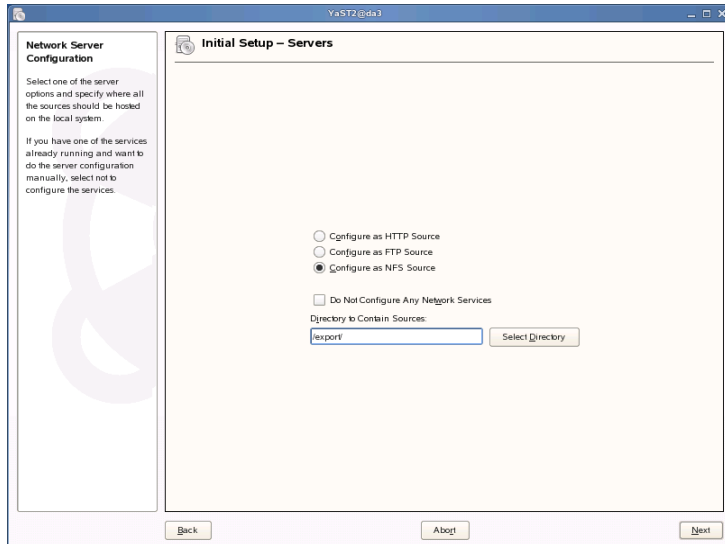
or enter

yast2 instserver

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

2. In the next dialog, select the protocol to access the installation server and the directory which will contain the installation repository:

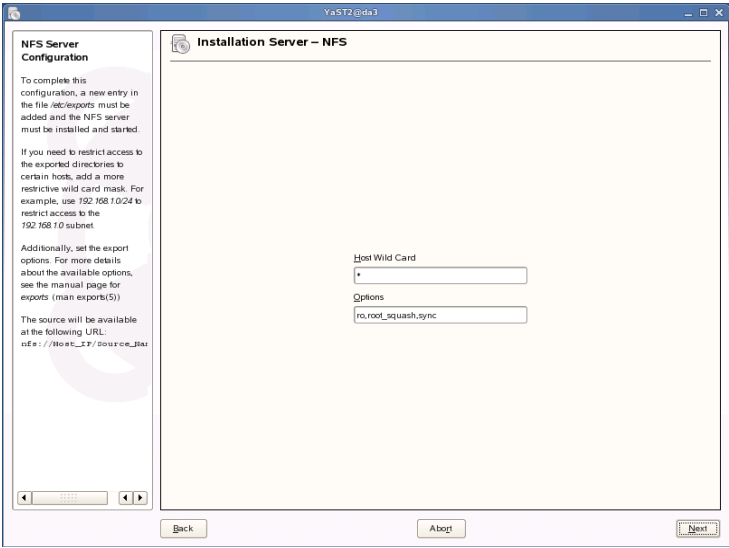
Figure A-2



We choose NFS for the protocol here.

3. Depending on the selected protocol, you have to configure the server. For NFS, you enter the values which will be written to the file `/etc/exports`:

Figure A-3



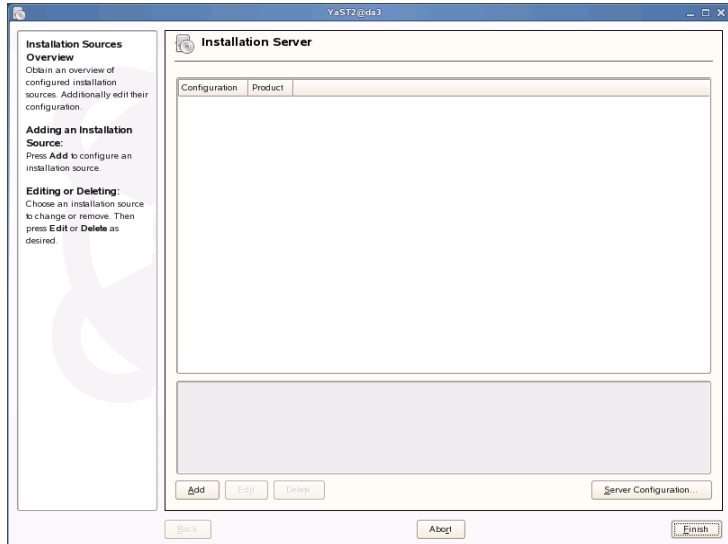
Accepting the default values here is recommended.



To get information about further options, enter in a terminal window **man 5 exports**.

4. After having configured the service, the following dialog appears:

Figure A-4

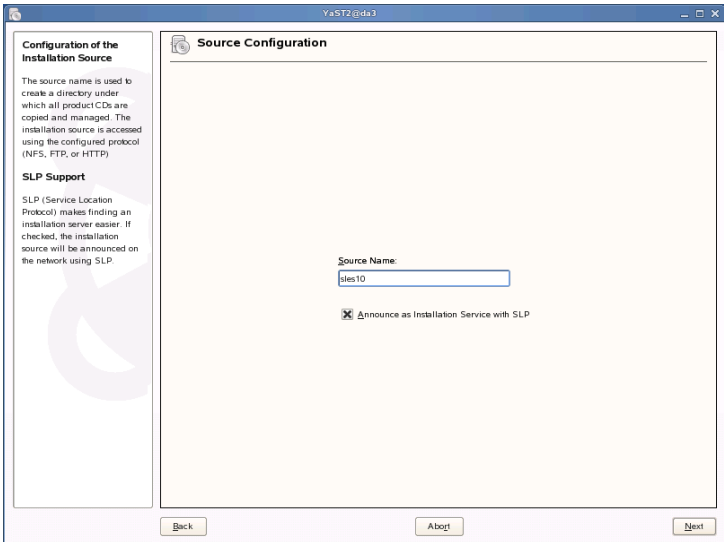


This is the first dialog displayed on any subsequent run of the YaST module “Installation Server”.

5. To fill an installation repository with information required for an installation, select **Add**.

6. Now you have to define the name of your source. This name will be used to create a directory which contains all the data for your installation server:

Figure A-5



By default, the installation CDs are used to copy all the files into the repository. If you select **Use ISO Images instead of CDs**, you have to provide the name of a directory which contains the ISO images.

You may also announce the installation server via SLP (Service Location Protocol) in the network.

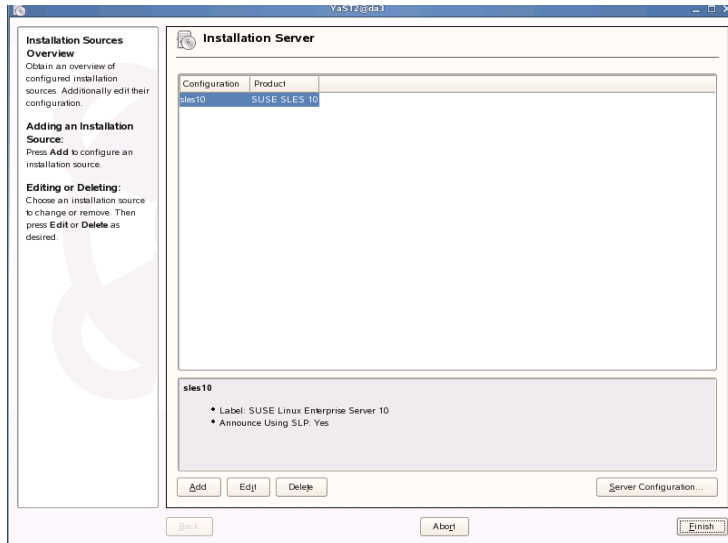
7. To start the creation of the installation repository, select **Next**.
The content of the CDs is copied into separate directories.



The installation server requires about 4 GB of disk space.

8. After copying the data into the repository, the following dialog is shown:

Figure A-6



9. To save the configuration, select **Finish**.

Objective 3 Create a Configuration File for AutoYaST

The easiest way to create a configuration file for AutoYaST is to use the YaST Autoinstallation module. Enter

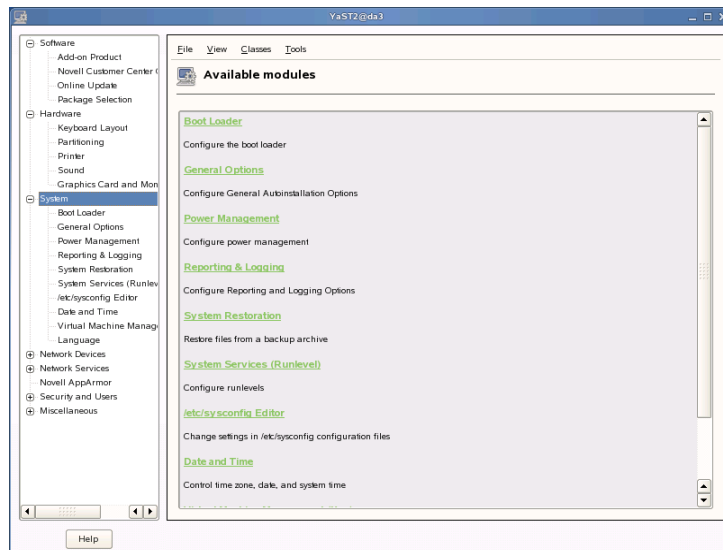
yast2 > Miscellaneous > Autoinstallation

or

yast2 autoyast

This module starts with the following dialog:

Figure A-7



The left window contains all parameters which can be configured in a tree-like structure.

By selecting a main entry, all available modules of this entry are displayed in the main window.

To select a module, either select the link in the main window or the name of the module in the left window.



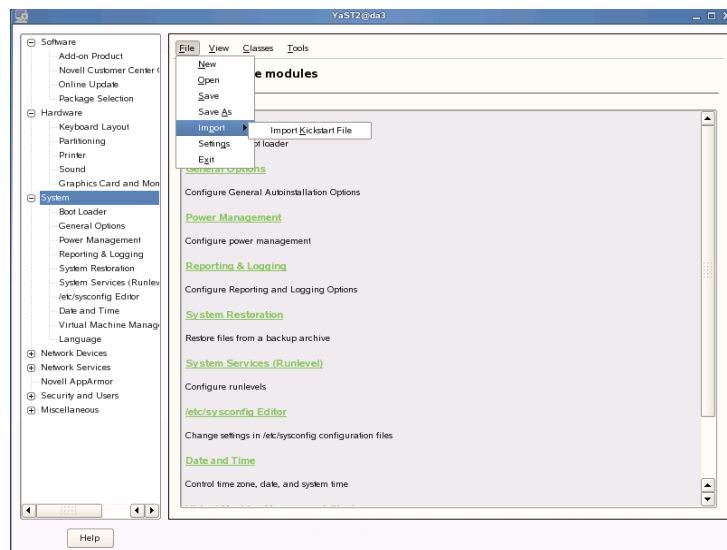
It is not necessary to configure every single aspect of the machines to be installed, as the automated installation makes use of the hardware detection capabilities of YaST. For instance, it is not necessary to provide the type of network card, the hardware detection will take care of this.

The main window displays the selected entry from the left window. Additionally, there is a menu bar at the top of this window.

In the **File** menu, you may open an existing AutoYaST configuration file or you may save your settings to a file.

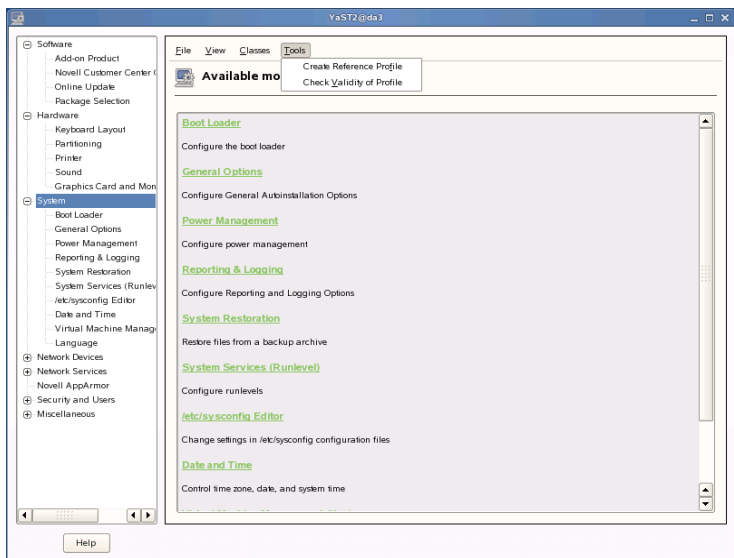
It is also possible to import Redhat Kickstart configuration files:

Figure A-8



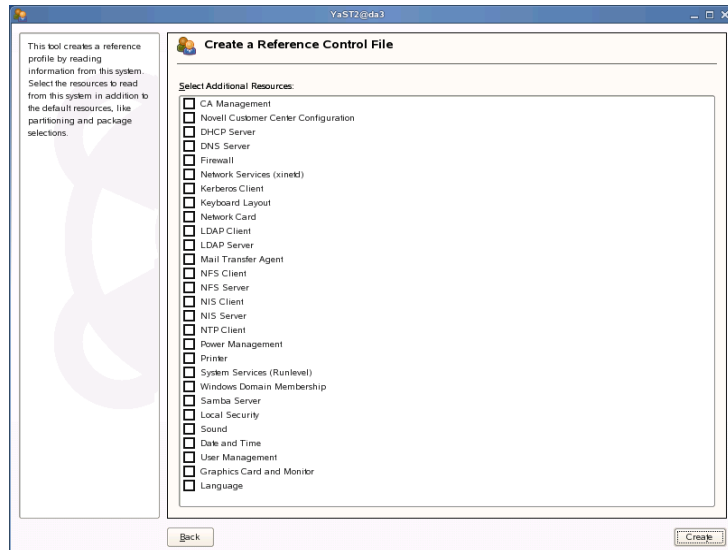
The **Tools** menu allows to use the current machine as a template. Select **Create Reference Profile**:

Figure A-9



The following dialog is shown:

Figure A-10



The reference profile is created by reading information from this system. By default, an exact copy of the configuration for all basic resources is created.

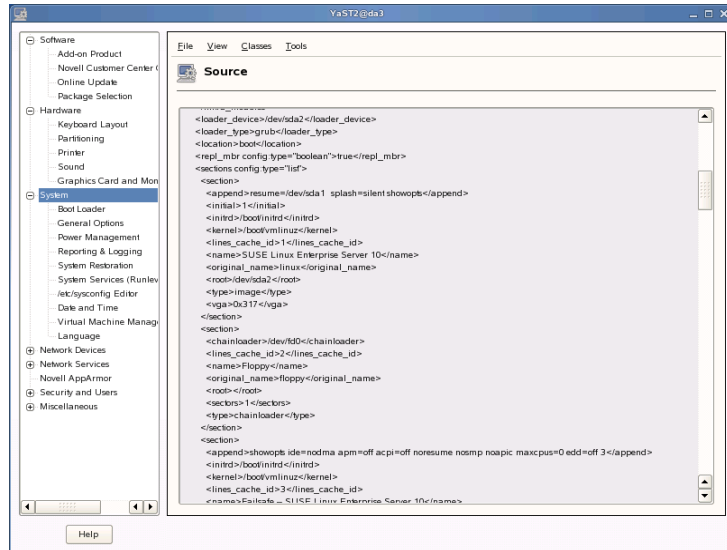
Add other necessary information for your machine using the check boxes in the main window.



Be sure to examine any resulting control file carefully before using it to autainstall a new system.

To view the configuration created, select **View > Source**:

Figure A-11



After you have finished your configuration, save it by selecting **File > Save as**.

A dialog box is shown with the default directory for AutoYaST configuration files, `/var/lib/autoinstall/repository/`. Enter a name for the file, e.g., `hostname.xml`.

You may change this via the Preferences menu.

Objective 4 Start the Installation

To start the automated installation, you need to copy the XML file generated to a floppy disk containing a FAT file system format.



Do not use a floppy disk with Ext2 file system format.

The easiest way of installing is using a file name of `autoinst.xml` on the floppy disk. If you saved your control file with a name of `da4.xml`, copy it to the floppy disk by entering

```
cp /var/lib/autoinstall/repository/da4.xml  
/media/floppy/autoinst.xml
```

Alternatively, you may enter

```
mcopyp /var/lib/autoinstall/repository/da4.xml  
a:autoinst.xml
```

The simplest way of installation is to have a DHCP server running which provides all network information during the installation.

1. Insert CD 1 and the floppy disk into your machine and start the boot process.
2. On the first boot screen, stop the boot process by pressing **Tab**.
3. Select **Installation**.
4. Now you have to provide the information that you want to use AutoYaST.

At the boot prompt, enter the following parameters (we assume here that the installation repository is available via NFS from 10.0.0.3/data/SLES10/):

```
autoyast=floppy:///autoinst.xml  
install=nfs://10.0.0.3/data/SLES10/  
splash=verbose
```

The last parameter switches to the detailed display during the boot process, so you can easily look at the boot messages.

After a short time, YaST is started. The installation starts with the Novell Software License Agreement (which is accepted automatically). After some checks, the packages are copied from the NFS server.

The system is rebooted at the end of the installation process. After the reboot, you may log in as root without a password if no password was set in the AutoYaST configuration file. You should immediately set a password for root.

Summary

Objective	Summary
1. Autoinstallation Basics	<p>AutoYaST is a tool for automated installation.</p> <p>The installation is controlled by an XML file which contains the machine specific information.</p> <p>AutoYaST is optimally used in conjunction with an installation server also running a TFTP and a DHCP server.</p> <p>The installation server can be accessed via the protocols NFS, HTTP and FTP.</p>
2. Set up an Installation Server	<p>To set up the installation server, use the YaST module “Installation Server”:</p> <p>yast2 > Miscellaneous > Installation Server</p> <p>or</p> <p>yast2 instserver</p>
3. Create a Configuration File for AutoYaST	<p>To create a configuration file for AutoYaST, use the YaST module “Autoinstallation”:</p> <p>yast2 > Miscellaneous > Autoinstallation</p> <p>or</p> <p>yast2 autoyast</p> <p>The default directory for AutoYaST configuration files is /var/lib/autoinstall/repository/.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Objective	Summary
4. Start the Installation	<p>To start the automated installation, copy the XML file to a floppy disk containing a FAT file system format.</p> <p>A DHCP server which provides all network information and an installation server simplifies the installation.</p> <p>Before starting the boot process, insert CD 1 and the floppy disk into the machine.</p> <p>You can also distribute the XML file via the network.</p>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

SECTION B **Appendix: Migrating Services from RedHat to SUSE Linux Enterprise Server 10**

The slides on the following pages are from a presentation held at Brainshare 06, covering different points to watch when migrating from RedHat to SUSE Linux Enterprise Server 10.

These slides were created prior to the launch of SLES 10. While some information might have become outdated, they nevertheless contain many tips you might find useful.

Objectives

1. Migrating Services

Objective 1 Migrating Services

Figure B-1



Novell.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-2

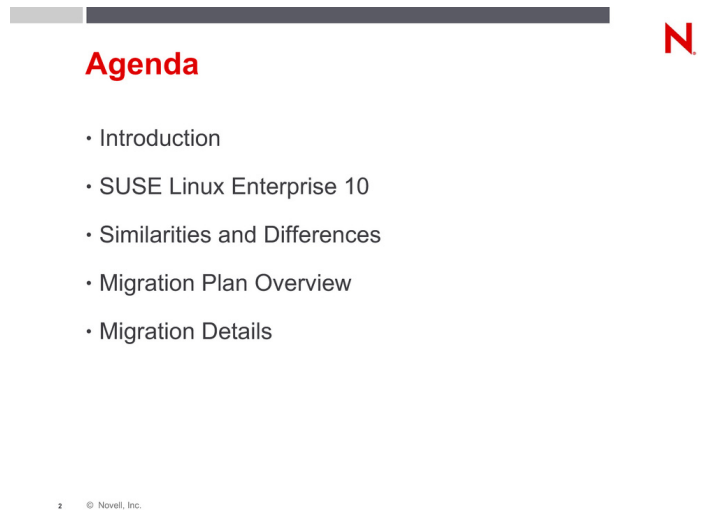


Figure B-3

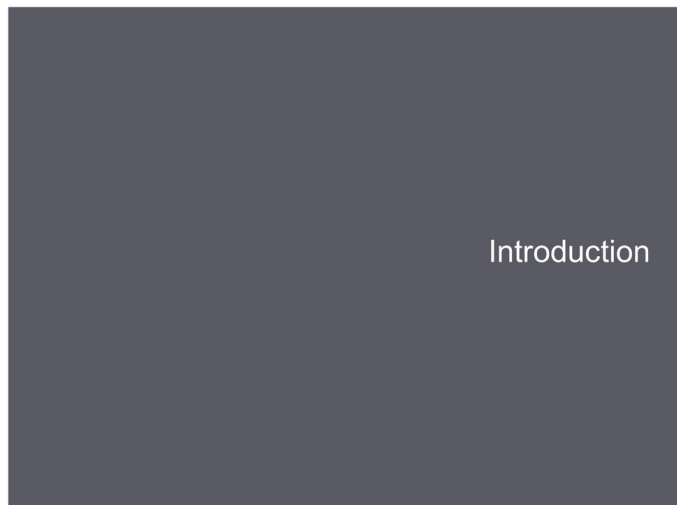


Figure B-4

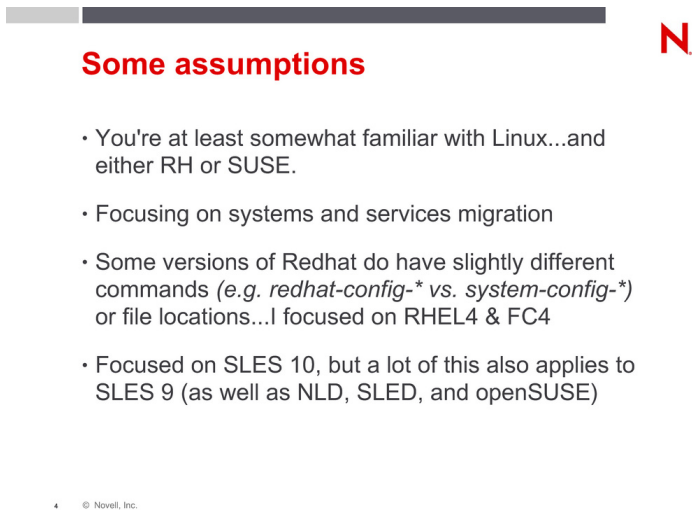


Figure B-5



Figure B-6

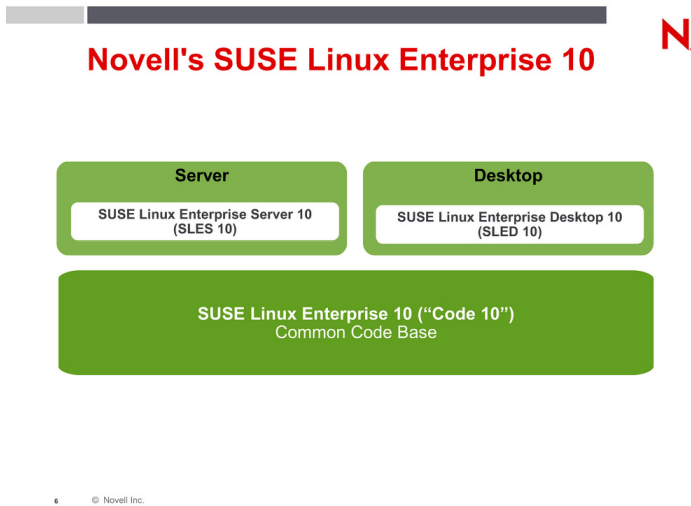
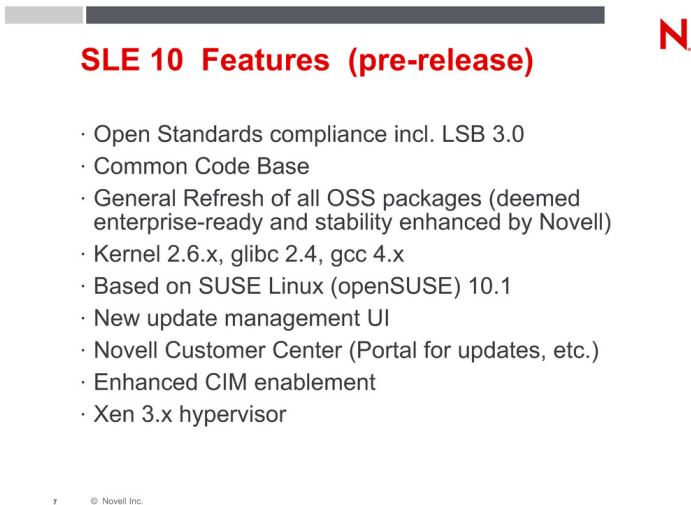


Figure B-7



1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-8

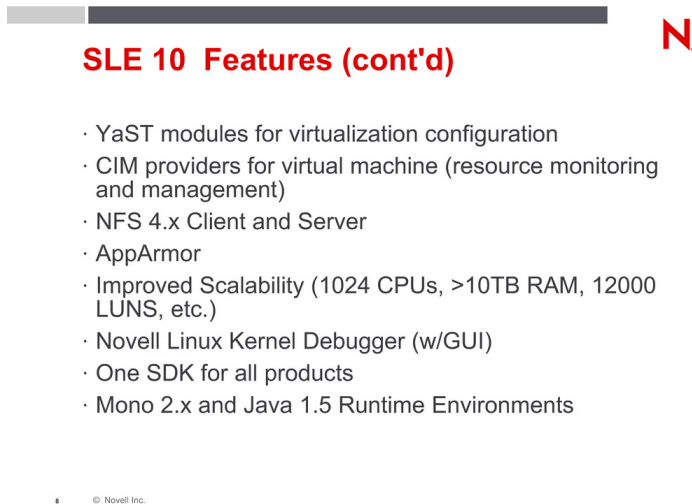


Figure B-9

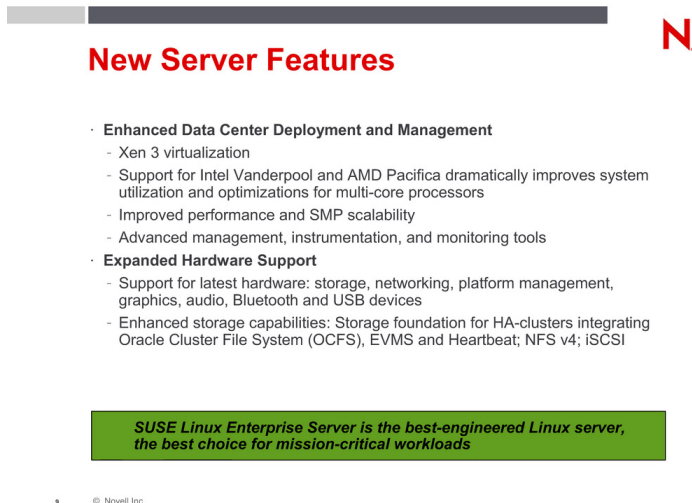



Figure B-10




New Desktop Features

- **New Levels of End User Productivity**
 - Powerful desktop productivity tools: E-mail, desktop search, Firefox Web browser, Instant messaging, multi-media
 - OpenOffice.org 2.0 office suite and Visual Basic macro conversion
 - Easy-to-learn desktop environment with advanced animations and rendering
- **Seamless Integration with Enterprise Systems**
 - Enhanced hardware support for USB and Bluetooth devices
 - Seamless connectivity to mixed networks, including Novell eDirectory and Active Directory
 - Printer support including auto driver download and installation, MS printer support
- **Flexible Deployment Options**
 - Thin and thick-client deployment
 - Easy deployment and desktop lock-down

Novell is #1 on the Linux desktop, providing vendor choice and dramatic costs savings for basic office workers

10 © Novell Inc.

Figure B-11



Novell Customer Center

- **Subscription and update service**
 - One tool to manage purchase, renewal, expirations.
 - Easy reporting on compliance and forecast future purchasing needs
 - Integrated patch management and update service, powered by ZENworks
- **Access to Novell Support**
 - Easy-to-use portal for support
 - One location for access to knowledge base, electronic support requests, and downloads
- **Intuitive management console**
 - One console to discover, update and manage Linux systems


Novell Network makes it easy to register, update, secure and monitor all Linux systems on the network, from the desktop to the data center

11 © Novell, Inc.


Figure B-12



Figure B-13



Some Differences



- Licensing/EULA
 - Subscription vs. License (and perpetual right-to-use)
- Separation/Inclusion of Support (*although not so true anymore*)
- Approach of Blended model vs. Pure open-source
- Vendor Lock-in
- Common code base/Modified kernel versions
- Partner Relationships/ISV Certification
- Package names and sometimes base version levels
- Included software/Add-ons
 - J2EE—JOnAS/Tomcat/JBoss; GFS/OCFS2; HA software

13 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-14

Some Differences (cont'd)

N

- Application Security (SELinux and AppArmor)
 - AppArmor has lower overhead, doesn't require program modifications, and is easy to use.
- Admin Tools
 - `system-config-*` vs. `yast`
- Update methods

14 © Novell, Inc.

Figure B-15

Some Similarities

N

- Actually more alike technically than different
- Share a ton of the same code...
- Both RPM-based distros
- LSB/FSH helps out here...know about LSB?
 - <http://www.linuxbase.org> - check it out!
- Use of `chkconfig` (for setting services per run level) and have start scripts in `/etc/init.d/*`
- Same init process and both use grub

15 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-16



Figure B-17

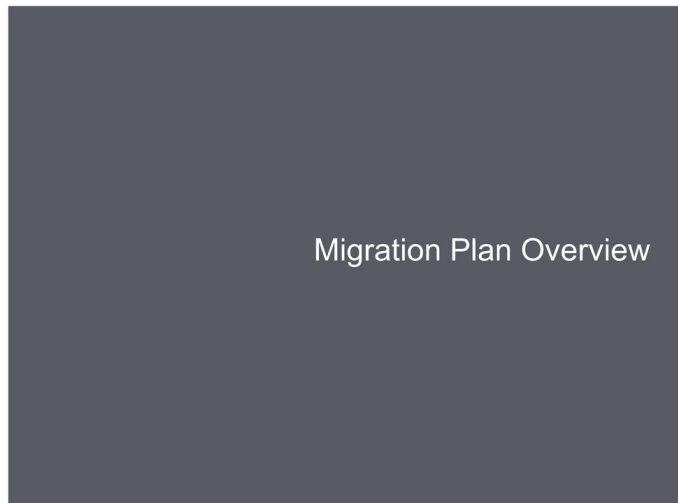
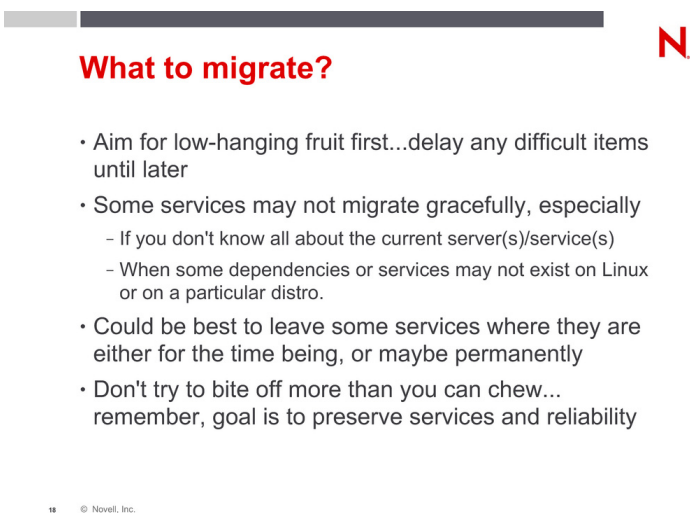
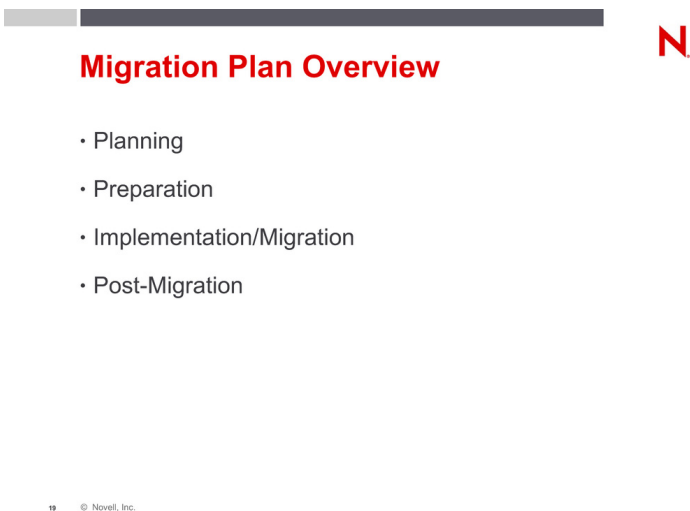


Figure B-18The slide features a title bar with a grey-to-dark-grey gradient. The title "What to migrate?" is in red. A large red "N" is in the top right. The content is a bulleted list. At the bottom left, there is a small number "18" and a copyright notice "© Novell, Inc.".

What to migrate?

- Aim for low-hanging fruit first...delay any difficult items until later
- Some services may not migrate gracefully, especially
 - If you don't know all about the current server(s)/service(s)
 - When some dependencies or services may not exist on Linux or on a particular distro.
- Could be best to leave some services where they are either for the time being, or maybe permanently
- Don't try to bite off more than you can chew...remember, goal is to preserve services and reliability

18 © Novell, Inc.

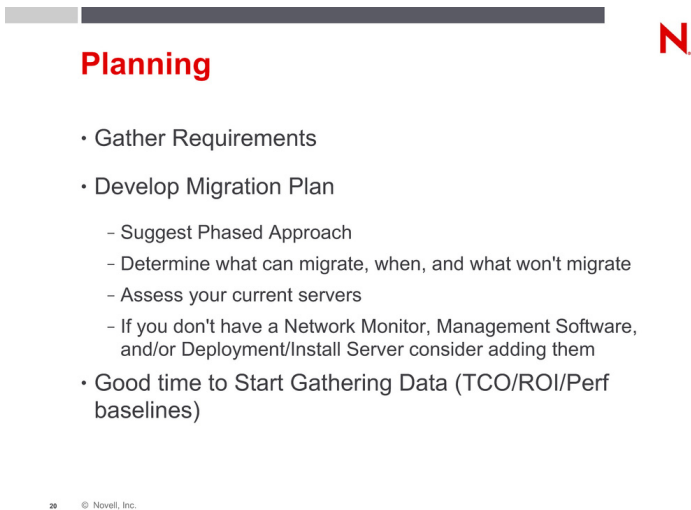
Figure B-19The slide features a title bar with a grey-to-dark-grey gradient. The title "Migration Plan Overview" is in red. A large red "N" is in the top right. The content is a bulleted list. At the bottom left, there is a small number "19" and a copyright notice "© Novell, Inc.".

Migration Plan Overview

- Planning
- Preparation
- Implementation/Migration
- Post-Migration

19 © Novell, Inc.

Figure B-20

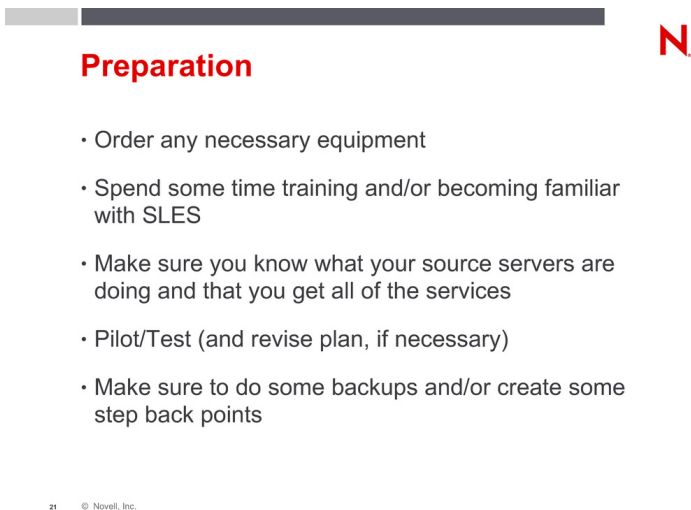


Planning

- Gather Requirements
- Develop Migration Plan
 - Suggest Phased Approach
 - Determine what can migrate, when, and what won't migrate
 - Assess your current servers
 - If you don't have a Network Monitor, Management Software, and/or Deployment/Install Server consider adding them
- Good time to Start Gathering Data (TCO/ROI/Perf baselines)

20 © Novell, Inc.

Figure B-21



Preparation

- Order any necessary equipment
- Spend some time training and/or becoming familiar with SLES
- Make sure you know what your source servers are doing and that you get all of the services
- Pilot/Test (and revise plan, if necessary)
- Make sure to do some backups and/or create some step back points

21 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-22

Implementation

N

- Build in parallel, if you can
- Test the servers after they are built (pre-prod)
- As you start trickling users/services over, check the logs...verify things are working as they should be
- Unplug the older boxes from the network, but leave them powered on for a bit (if you can)
- BEST PRACTICES: phased approach, keep it simple, check/test often, and cover your bases

22 © Novell, Inc.**Figure B-23**

Post-Migration

N

- Finalize TCO/ROI Data (if gathering data)
- Do some performance checks (compare against earlier baselines)
- Don't forget to document what you did, especially if you ran into problems...it might save you some time down the road
- Determine how you can refine the process for next migration
- Consider some tuning

23 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-24

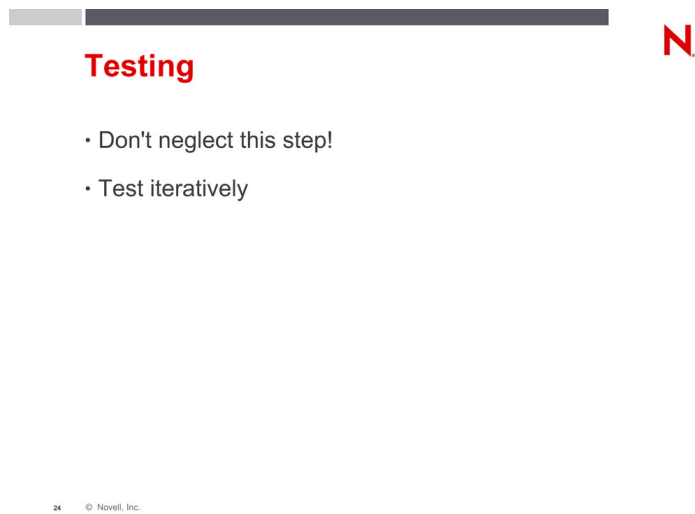


Figure B-25

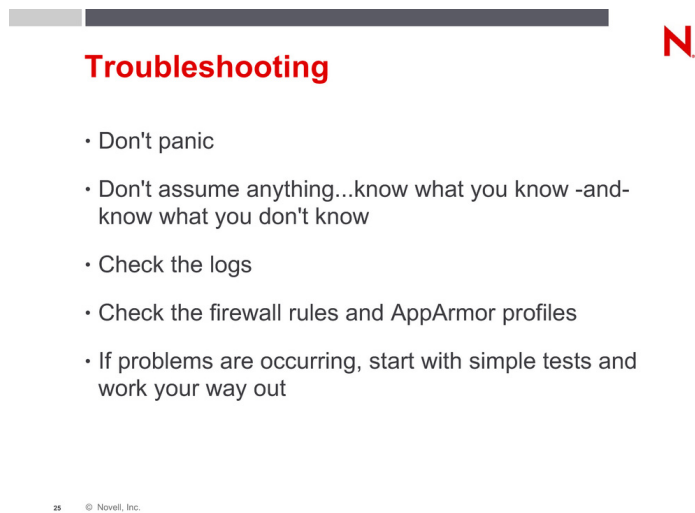


Figure B-26

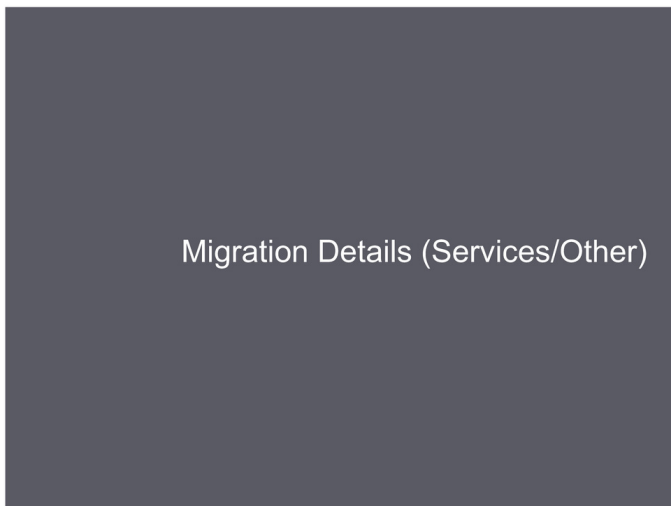
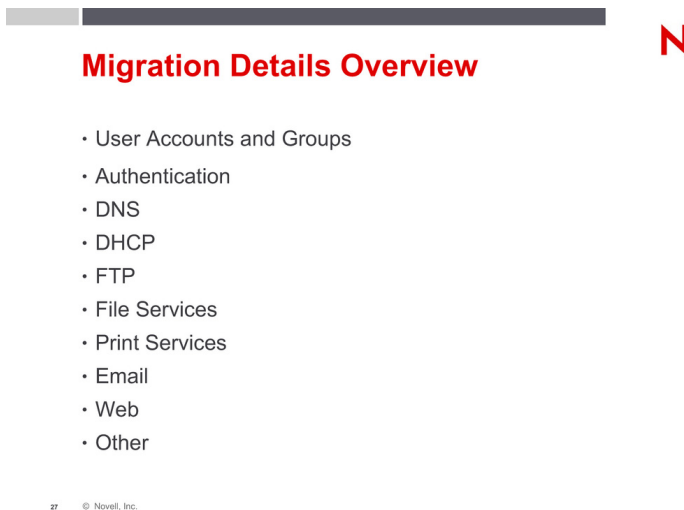


Figure B-27



1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-28

A few tips and tricks...

N

- Try to go the route of the distro...
- SuSEconfig - For the services, check `/etc/sysconfig/*` and modify values there, and then run SuSEconfig... it'll help you with changes to the config files
- If you like to hand-edit files, you can turn off SuSEconfig, so it doesn't clobber your files
 - edit `/etc/sysconfig/suseconfig`, change value for `ENABLE_SUSECONFIG` to "no"
 - run `SuSEconfig`

28 © Novell, Inc.

Figure B-29

User Accounts/Groups

N

File(s)/File locations:

- | | |
|---|-----------------------------------|
| - Main user/password file (local accts) | <code>/etc/passwd</code> |
| - Shadow password file (passwords are here) | <code>/etc/shadow</code> |
| - Group file | <code>/etc/group</code> |
| - Default login settings | <code>/etc/login.defs</code> |
| - Default settings for useradd tool | <code>/etc/default/useradd</code> |
| - Default user directory source | <code>/etc/skel</code> |

Configure (GUI/CUI):

- **RHEL/Fedora:** `system-config-users`
- **SLES/SUSE:** `yast` (Security and Users -> User Management)
`yast users`

Notes:

- Some user dotfiles won't be checked (e.g. `.xinitrc`, `.bashrc`, etc.); you can either modify the way SLES does it or modify your scripts a bit

29 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-30

User Accounts/Groups (cont'd)

N

Notes (cont'd):

- RH and SUSE encrypt their passwords using different methods...you could either modify SLES/SUSE to use the same encryption method or you can just recreate the passwords on the new system. See next slide for more info
- Regular user IDs start at 1000 on SUSE; 500 on RH (by default)
- RH creates user-specific groups and places user in that group; SUSE places users in the same group, users.
- Some minor UIDs/GIDs do differ between the two (mainly system accounts)...'nobody' is the one that caught me
- Might suggest recreation of accounts via script...and good time to reset passwords too
- Consider an external directory (NIS/NIS+/LDAP/eDirectory/etc.) for user accounts as opposed to local user accounts if you can. Might check out <http://www.padl.com/OSS/MigrationTools.html> for migrate_passwd.pl (for passwd->LDIF); other tools exist as well.
- Test with a test user account to ensure transition is smooth

30 © Novell, Inc.

Figure B-31

Authentication (and PAM)

N

File(s)/File locations:

- Config stublet files directory /etc/pam.d/*
- (usually ignored if above is present) /etc/pam.conf

Configure (GUI/CUI):

- RHEL/Fedora: `system-config-authentication`
- SLES/SUSE: `yast (Security and Users -> Local Security)`
`yast security`

Notes:

- RHEL and SUSE encrypt their passwords differently...so encrypted passwords won't transfer straight over. You can modify PAM on SLES/SUSE to use the same method as RH if you want. You'd have to modify /etc/pam.d/passwd – for more info, see <http://lists.suse.com/archive/suse-linux-e/2000-Dec/0867.html>

31 © Novell, Inc.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure B-32

DNS

File(s)/File locations:

- Start script /etc/init.d/named
- Main config file /etc/named.conf
- Zone files (RH) /var/named/*
- Zone files (SUSE) /var/lib/named/*
- RNDK Key file (for remote DNS admin) /etc/rndc.key
- RNDK config file section (RH only) /etc/rndc.conf

Configure (GUI/CUI):

- n/a—by hand or non-default admin tool (e.g. webmin)

Notes:

- Pretty similar between the two distros...zone files should be able to transfer just fine. Check your paths, locations, etc. to be sure.
- bind-chroot packages will push zone files down a couple of levels (e.g. /var/named/chroot/var/named/*)

N

32
© Novell, Inc.

Figure B-33

DNS (cont'd)

Notes (continued):

- Another method outside of a straight copy of the files, is to set the new server up as a secondary and then have them swap roles later. Never hurts to have another DNS around.
- You can also use other products to host DNS (e.g. eDirectory or LDAP).

N

33
© Novell, Inc.

Figure B-34

N

NIS/yp

File(s)/File locations:

- Start script /etc/init.d/ypbind
- Config file /etc/yp.conf
- Data dir/files /var/yp/*

Configure (GUI/CUI):

- RHEL/Fedora: n/a
- SLES/SUSE: yast (Network Services -> NIS Server)
yast nis_server

Notes:

- Really nothing of significance to note...should go quite smoothly.
- Suggest start by making it a secondary.

34

© Novell, Inc.

Figure B-35

N

DHCP

File(s)/File locations:

- Start script /etc/init.d/dhcpd
- Main config file /etc/dhcpd.conf
- Files directory /var/lib/dhcp

Configure (GUI/CUI):

- RHEL/Fedora: n/a
- SLES/SUSE: yast (Network Services -> DHCP Server)
yast dhcp-server

Notes:

- SLES/SUSE can be integrated with LDAP

35

© Novell, Inc.

Figure B-38

N

File Services—Samba

File(s)/File locations:

- Start scripts (RH) `/etc/init.d/smb`
- Start scripts (SUSE) `/etc/init.d/winbind`
- Config files `/etc/init.d/smb`
`/etc/init.d/nmb`
`/etc/init.d/winbind`
`/etc/samba/*`

Configure (GUI/CUI):

- RHEL/Fedora: `system-config-samba`
- SLES/SUSE: `yast (Network Services -> Samba Server)`
`yast samba-server`

Notes:

- Similar versions...nothing significant to note
- Yast has another module for samba client portion

38

© Novell, Inc.

Figure B-39

N

File Services—NFS (server)

File(s)/File locations:

- Start script (requires nfsboot and portmap) `/etc/init.d/nfs`
- Main exports file `/etc/exports`

Configure (GUI/CUI):

- RHEL/Fedora: `system-config-nfs`
- SLES/SUSE: `yast (Network Services -> NFS Server)`
`yast nfs-server`

Notes:

- Nothing significant to note

39

© Novell, Inc.

Figure B-40

Print Services

File(s)/File locations:

- Start script `/etc/init.d/cups`
- Config files `/etc/cups/*`

Configure (GUI/CUI):

- RHEL/Fedora: `system-config-printer`
- SLES/SUSE: `yast (Hardware -> Printer)`
`yast printer`

Notes:

- Both distros use cups; can load LPRng if so desired
- Older versions of RH might be using LPRng

40 © Novell, Inc.

Figure B-41

E-mail (postfix/sendmail)

File(s)/File locations:

- Start script `/etc/init.d/postfix`
- Config files `/etc/postfix/*`
- Mail queue `/var/spool/postfix`

Configure (GUI/CUI):

- RHEL/Fedora: `system-config-mail`
- SLES/SUSE: `yast (Network Services -> Mail Transfer Agent)`
`yast mail`

Notes:

- SUSE runs postfix by default; can load sendmail if so desired
- Test, test, test (especially if changing from sendmail -> postfix)

41 © Novell, Inc.

Figure B-42

N

Web (apache)

File(s)/File locations:

- Start script (RH) /etc/init.d/httpd
- Start script (SUSE) /etc/init.d/apache2
- Config files (under conf and conf.d) /etc/httpd/*
- Config files /etc/apache2/*
- Files directory/wwwroot (RH) /var/www/html
- Files directory/wwwroot (SUSE) /srv/www/htdocs

Configure (GUI/CUI):

- RHEL/Fedora: system-config-httpd
- SLES/SUSE: yast (Network Services -> HTTP Server)
yast http-server

Notes:

- SLES/SUSE has a unique structure...consult the httpd.conf. Makes use of config stublet files...much more manageable than a single large httpd.conf

42

© Novell, Inc.

Figure B-43

N

SSH

File(s)/File locations:

- Start script /etc/init.d/sshd
- Config files /etc/ssh/*

Configure (GUI/CUI):

- RHEL/Fedora: n/a
- SLES/SUSE: n/a

Notes:

- Default firewall rules on SLES block incoming ssh

43

© Novell, Inc.

Figure B-44

N

Remote Admin (VNC)

File(s)/File locations:

- Start script (RH) `/etc/init.d/vncserver`
- Config files (RH) `/etc/sysconfig/vncservers`
- Start/Config files (SUSE) `/etc/xinetd.d/vnc`

Configure (GUI/CUI):

- RHEL/Fedora: `n/a`
- SLES/SUSE: `yast (Network Services -> Remote Administration)`
`yast remote`

Notes:

- RH - `vnc-server`; SUSE - `tightvnc`
- SLES/SUSE - `vnc` provided via `xinetd`; SLES/SUSE service provides both web and `vnc` access (diff ports)
- Default firewall rules on block incoming unless changed subsequently (via VNC/Remote Admin setup or manually)

44 © Novell, Inc.

Figure B-45

N

Other

- Other Services (e.g. NTP, NIS/yp, MySQL, etc.)
- Boot Options
 - kernel options
 - installation media/server options (ftp/http/nfs/local media)
- Kickstart vs. AutoYaST
- Update methods (yum, up2date/rhn, rug/zlmserver, YOU, other)

45 © Novell, Inc.

SECTION C Appendix: A Guide to SUSE Linux Enterprise Server for Red Hat Users

This guide was written in 2005 and focuses on SUSE Linux Enterprise Server 9. Some information in this guide does not apply to SUSE Linux Enterprise Server 10, however most of it remains valid.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-1

A Guide to SUSE™ Linux Enterprise Server

For Red Hat® Users

www.novell.com

NOVEMBER 2005



Novell.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-2

Disclaimer	Novell, Inc. makes no representations or warranties with respect to the contents or use of this document and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose.	
Trademarks	Novell, the Novell logo, the N logo and ZENworks are registered trademarks; SUSE is a trademark; and Premium Service is a service mark of Novell, Inc. in the United States and other countries. *Red Hat is a registered trademark of Red Hat, inc. Linux is a registered trademark of Linux Torvalds. Java and J2EE are registered trademarks of Sun Microsystems, Inc. IBM is a registered trademark of IBM Corporation. Microsoft and Windows are registered trademarks of Microsoft Corporation. All other third-party trademarks are property of their respective owners.	
	© 2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system or transmitted without the express written consent of Novell, Inc.	
	Novell, Inc. 1800 South Novell Place Provo, UT 84606 USA	
Prepared By	Ross Brunson, Ron Terry, Jason Ganovsky and Justin Steinman	

Figure C-3

Overview	5
Audience	5
Red Hat Enterprise Linux	5
Novell Enterprise Linux	5
Topics Covered	6
Installation	6
Overview of Automated Installations	6
Install Modes	9
Boot Process and Runlevels	11
Boot Loader Differences	11
Init scripts	11
The SUSE Linux Enterprise Server Boot Process	12
Init and Runlevels	13
SUSE Linux Enterprise Server Boot Options	13
Services and Daemons	15
Network Configuration	18
User Management and Environment	18
Command-line User Management	18
Graphical User Management	19
User Environment	20
System Administration	21
SuSEconfig—The Normalizer	21
YaST The GUI Version	21
YaST— the Text Mode Version	22
YaST—the Text-mode Version	23
Comparing System Administration Tools	23
Service and Support Pack vs. Red Hat Updates	24
Enabling Red Hat Network Support	25
Enabling SUSE Linux Enterprise Server Support	25

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-4

4

File-system Hierarchy	25
Documentation and Help Resources.....	26

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-5

6

OVERVIEW

This white paper assists IT administrators who have to replace their Red Hat® Enterprise Linux servers with SUSE™ Linux Enterprise Server. Distributed by Novell®, SUSE Linux Enterprise Server empowers businesses to leverage Linux® and open source by delivering a scalable, high-performance foundation for secure enterprise computing. This white paper covers the technical needs for integrating SUSE Linux Enterprise Server into an existing IT environment, with particular emphasis on replacing legacy Red Hat Linux systems with SUSE Linux Enterprise Server systems.

Both the Red Hat and the SUSE Linux distributions are similar in focus and scope. However, enough differences exist to make any integrations or migrations frustrating unless a mapping is successfully made so that system administrators know where both distributions store the same configuration file or how they perform the same task.

AUDIENCE

This white paper targets the professional system administrators who are currently running Red Hat 5.x-9.x or Red Hat Enterprise Linux Version 3 and beyond. The format is designed to help Red Hat system administrators find the corresponding SUSE Linux Enterprise Server topic and provide a side-by-side comparison and set of steps for configuring the SUSE Linux Enterprise Server system.

RED HAT ENTERPRISE LINUX

Red Hat Enterprise Linux 4 is a family of Linux server and desktop operating systems. Delivered to market in early 2005, Red Hat Enterprise Linux 4 adopts the latest Linux kernel 2.6 technology and its benefits: better performance, scalability, security and hardware support. When purchasing Red Hat Enterprise Linux 4, customers gain access to Red Hat Network, a subscription-service bundle that keeps software up to date and provides access to Red Hat technical support. The Red Hat Enterprise Linux 4 product family includes the AS version for advanced servers, the ES version for mid-range servers and the WS version for desktop users.

Red Hat offers clustering software separately, and the company plans to offer new directory server and Java® 2 Enterprise (J2EE®) application server. For more information, visit: www.redhat.com

NOVELL ENTERPRISE LINUX

Novell offers enterprise Linux servers and desktops based on the latest kernel 2.6. Shipping since August 2004, SUSE LINUX Enterprise Server 9 supports native kernel 2.6 features and is ideal for application hosting, high-performance computing and Web-infrastructure workloads.

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-6

6

Novell offers other products based on SUSE LINUX Enterprise Server, including Novell Linux Desktop 9, an enterprise-grade workstation operating system and application suite, and Novell Open Enterprise Server, which brings popular Novell file, print, directory, clustering and identity-management solutions to Linux. Novell ZENworks® Linux Management, available separately, provides inside-the-firewall management of Linux software. For more information, visit: www.novell.com/linux

TOPICS COVERED

The remainder of this white paper covers the technical differences between Red Hat Enterprise Linux and SUSE Linux Enterprise Server, including:

- Installation
- Boot processes and run levels
- Services and daemons
- User management and environment
- System administration
- Service and support packs vs. Red Hat updates
- File-system hierarchy
- Documentation and help resources

INSTALLATION

While both Red Hat Enterprise Linux 4 and SUSE Linux Enterprise Server 9 feature GUI and text-based manual installation methods *and* automated installation methods, they take somewhat different approaches. Red Hat Enterprise Linux uses the Anaconda program to guide the installation; SUSE Linux Enterprise Server uses YaST. Both employ various sub-modules for tasks such as partitioning and configuring X.

Additionally, both include methods for installing systems in completely hands-off mode; Red Hat's is called Kickstart, and Novell offers AutoYaST. The concept of an across-the-network or automated installation is not new to the system administrator's world, and although the systems take a slightly different approach, they both need the installation sources (packages and files), an answer file to provide the configuration for the target installation and the various methods of getting the target machine booted and in contact with the required resources.

Overview of Automated Installations

In the sections that follow, we compare tasks in the two installations: Red Hat Enterprise Linux on the left and SUSE Linux Enterprise Server on the right. In cases of disparate tasks, linkage is provided.

Note: When comparing the two systems, Red Hat's Kickstart can appear simpler due to the number of steps that an action takes, but AutoYaST makes up for a slightly more complex setup with its plethora of options and extreme flexibility.

Figure C-7

7

Step 1	
Steps for an Automated Installation (Both Distributions):	
1. Create an installation answer file	
2. Make the installation source available	
3. Make the installation answer file available (CD or network)	
4. Boot and start the installation	

Step 2	
Make the Installation Source Available	
Red Hat Enterprise Linux	SUSE Linux Enterprise Server
1. Make a directory to store the installation source. For example <code>/export/install</code>	1. As the root user, create the directory <code>/export/sles9nfs</code>
2. Copy the Red Hat Enterprise Linux CD-ROMs either as a set of directories, or, as an easier option, copy the CD-ROMs as ISO files into the install directory	2. From the desktop, select <i>N -> System -> YaST -> Misc -> Installation Server</i>
3. Export the directory via NFS by editing the <code>/etc/exports</code> file and placing a line similar to the following as one of the lines in the file <code>/export/install *(ro)</code>	3. In the following screen check the box next to <i>Configure as NFS Source</i>
4. Restart the NFS Server via the service command <code>service nfs restart</code>	4. In the <i>Directory to contain sources</i> text box enter the text <code>/export/sles9nfs</code> and then click <i>Next</i>
5. Verify the share is properly shown on the network with the command <code>showmount -e localhost</code>	5. You will be prompted for the <i>Host Wild Card</i> and the <i>Options for NFS</i> . Keep the defaults and just click <i>Next</i>
6. The server is ready for duty as an installation server	6. You will then be presented with the <i>Source Configuration</i> screen
Note: While the number of steps needed to create a Red Hat NFS installation server are less than SUSE's comparative setup, the SUSE process has options for updating the Installation Server files with Support Packs and Additional Product CD's, something that's	7. Click the <i>Configure</i> button
	8. In the <i>Source Name</i> text field enter the text <code>/export/sles9nfs</code>
	9. Check the box next to <i>Use ISO Images Instead of CDs</i>
	10. In the <i>Directory with CD Images</i> text box, enter the directory path that contains the ISO

Figure C-8

Make the Installation Source Available	
more complex and less well guided on a Red Hat system.	<p>files and click <i>Next</i></p> <p>11. The <i>select file</i> dialog will show with the contents of your ISO directory. Click on the first ISO file and click <i>Open</i></p> <p>12. Repeat this process until all six ISOs have been read</p> <p>13. When presented with the <i>Source Configuration</i> screen again, click <i>Finish</i></p>

Step 3

Make the Installation Answer File Available	
Red Hat Enterprise Linux	SUSE Linux Enterprise Server
<p>1. Copy the isolinux directory from the Red Hat Enterprise Linux CD # 1 to /tmp/CDROOT</p> <p>2. Change to the /tmp/CDROOT directory</p> <p>3. Change the permissions on the files in /tmp/CDROOT/isolinux to be writable for the user owner</p> <p>4. Copy the control file (it must be named ks.cfg) to the /tmp/CDROOT/isolinux directory</p> <p>5. From the /tmp/CDROOT directory, make the bootable CD with the command:</p> <pre>mkisofs -o bootcd.iso -b isolinux.bin -c boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -R -J -v -T isolinux/</pre> <p>6. Burn the resulting bootcd.iso to a CD-ROM blank and test</p>	<p>1. Copy the /boot directory from the SUSE Linux Enterprise Server 9 CD1 CD-ROM to /tmp/CDROOT</p> <p>2. Change to the /tmp/CDROOT directory</p> <p>3. Change the permissions of the <i>isolinux.cfg</i> file to be writable for the user owner</p> <p>4. Copy the control file (it must be named <i>autoinst.xml</i>) to the /tmp/CDROOT/boot directory</p> <p>5. From the /tmp/CDROOT directory issue, make the bootable CD with the command:</p> <pre>mkisofs -o bootcd.iso -b isolinux.bin -c boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -R /tmp/CDROOT</pre> <p>6. Burn the resulting bootcd.iso to a CD-ROM blank and test</p>

Figure C-9

Step 4

Boot and Start the Installation	
Red Hat Enterprise Linux	SUSE Linux Enterprise Server
1. Boot with CD # 1 If the <i>ks.cfg</i> file is on a floppy, enter the following at the boot prompt: <i>linux ks=floppy</i> 2. Booting with the customized CD If the <i>ks.cfg</i> file is on the customized Boot CD, enter the following at the boot prompt: <i>linux ks=cdrom:/ks.cfg</i> 3. For NFS Installation, enter the following at the boot prompt: <i>ks=nfs:server:/dir/ks.cfg</i> 4. For HTTP installation, enter the following at the boot prompt: <i>ks=http://server/dir/ks.cfg</i> 5. Once you press <i>Enter</i> , the installation should begin and finish without any need for interaction	1. Boot with CD # 1, and when the CD menu shows, choose the Installation option 2. Press <i>F3</i> and choose the appropriate installation source, then fill in the server IP address and directory text boxes Note: Alternatively, you can enter <i>install=proto://server/dir</i> where <i>proto</i> is the NFS HTTP FTP protocol, <i>server</i> is the IP address of the server and <i>dir</i> is the directory where the packages reside 3. Press <i>F5</i> and choose the <i>Verbose</i> option to show all messages during the installation for troubleshooting purposes 4. In the <i>GRUB Boot Options</i> text box, enter one of the following: - For NFS installation: <i>autoyast=nfs://server/dir/autoinst.xml</i> - For HTTP installation: <i>autoyast=http://server/dir/autoinst.xml</i> - For disk-based installation: <i>autoyast=device://sda1/autoinst.xml</i> 6. Press <i>Enter</i> to execute the installation; no further interaction should be necessary

Install Modes

Not only do Red Hat Enterprise Linux and SUSE Linux Enterprise Server provide the ability to use different types of installation media for the source of the installation, but they also provide several different install modes (ways to perform the installation).

Typically, Linux installations are performed directly via the locally attached keyboard monitor and mouse. However, in some cases (such as on mainframes or headless servers), this may be impossible. Instead of relying on expensive third-party components such as IP KVMs or hardware remote-access cards, SUSE Linux Enterprise Server bundles remote installation methods such as

Figure C-10

10

VNC (Web- or client-based), SSH or even telnet. Although Red Hat Enterprise Linux does support those options, initiating these methods is more complex and much less clear than it is in SUSE Linux Enterprise Server. Any of these remote-access methods may be specified prior to installation at the boot prompt.

Using VNC

To enable a VNC-based installation on SUSE Linux Enterprise Server, simply add the following to the boot options when initiating the install:

```
vnc=1 vncpassword=password
```

A typical remote-installation boot options line that will use VNC will use a boot options line like the following:

```
insmod=pcnet32 vnc=1 vncpassword=novell dhcp=1
```

```
install=http://10.0.0.254/sles9/sles9inst
```

Note: The above command line options should be typed in the boot options dialog as one line.

The system will initiate the installation, enable the VNC server and supply the VNC server with the proper password. It will then display an IP address and server instance on the screen for you to attach to with a Java-enabled browser or a VNC client on any platform.

Using SSH

A similar process is used for an SSH installation. You can use the same boot options line as shown above, replacing the VNC-related information with SSH setup info:

```
insmod=pcnet32 usessh=1 password=novell dhcp=1
```

```
install=http://10.0.0.254/sles9/sles9inst
```

Note: The above command line options should be typed in the boot options dialog as one line.

Upon starting the installation binaries, the screen will display an IP address to which you can attach as the root user using your SSH client. This will cause the YaST installer to display on your local machine.

Note: When using SSH to connect and redirect X applications, use the `-X` parameter. For example, if the system being installed has the IP address 10.0.0.144, you would use the following SSH command to attach to it properly:

```
ssh root@10.0.0.144 yast
```

Figure C-11

Another parameter that you may have to use is the one that forces text mode for an installation. The installation routine defaults to a GUI X-based YaST display if it is possible, so to run in text mode, add the "textmode=1" parameter to the boot options dialog entries.

Note: A full layout of the SUSE Linux Enterprise Server boot options in included in a later section.

BOOT PROCESS AND RUNLEVELS

Boot Loader Differences

The default boot loader for SUSE Linux Enterprise Server is grub, and the file system layout for all related boot files is generally the same as in Red Hat Enterprise Linux. The one notable exception is while Red Hat Enterprise Linux maintains a link from `/boot/grub/menu.lst` to `/etc/grub.conf`, which is, in turn, a link back to `/boot/grub/grub.conf`, no such link exists in SUSE Linux Enterprise Server. In SUSE Linux Enterprise Server, the `/boot/grub/menu.lst` is a regular file and contains all grub boot menu settings, while `/etc/grub.conf` contains information needed by the grub command to install itself into the MBR.

Note: You'll see this two-stage process used throughout the SUSE Linux Enterprise Server system where a service's or command's environment configuration is separated from its service-oriented configuration; it makes more sense than jumbling all the configuration into a single file. This can be particularly helpful when troubleshooting processes that have complex configurations.

Init scripts

Both Red Hat Enterprise Linux and SUSE Linux Enterprise Server use similar init script structures such as the `/sbin/init` as the first process, the PID 1 assignment. Moreover, both use the `/etc/inittab` file for the runlevel definition and general housekeeping of what starts on boot during the runlevels and at various system events.

The following table compares the two systems' boot process layouts:

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Kernel	<code>/boot/vmlinuz-<version></code>	<code>/boot/vmlinuz</code> (symlink)
Initrd	<code>/boot/initrd-<version></code> uses the <code>/initrd</code> directory	<code>/boot/initrd</code> (symlink)
Grub	<code>/boot/grub/menu.lst</code> is a link to <code>/boot/grub/grub.conf</code> <code>/etc/grub.conf</code> is a symlink to <code>/boot/grub/grub.conf</code>	<code>/boot/grub/menu.lst</code> <code>/etc/grub.conf</code> (only for installation purposes)

Figure C-12

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Sysinit Scripts	<i>/etc/rc.d/rc.sysinit</i> <i>/etc/rc.d/rc.local</i>	<i>/etc/init.d/boot</i> <i>/etc/init.d/boot.*</i> <i>/etc/init.d/boot = runlevel</i>
Modules	<i>/etc/modprobe.conf</i> only lists aliases for boot time load <i>/etc/modprobe.conf.dist</i> lists all modules that can be loaded	<i>/etc/modprobe.conf</i> lists all modules that can be loaded <i>/etc/modprobe.d</i> (include directory for modprobe) <i>/etc/modprobe.conf.local</i>

Note: SUSE Linux Enterprise Server uses *<filename>.local* files for a lot of items. The use of a *.local* file ensures that when a security patch or fix for a bug comes out and the updated package is applied to the system, the *.local* file that matches up with the configuration file will not be touched; it is solely the realm of the system administrator.

The SUSE Linux Enterprise Server Boot Process

SUSE Linux Enterprise Server uses a boot process that is similar to the process employed by Red Hat Enterprise Linux in many ways, but while Red Hat Enterprise Linux uses the kudzu tool, SUSE Linux Enterprise Server uses the linuxrc script to do the system initialization tasks.

The following list explains the SUSE Linux Enterprise Server boot process in detail:

1. Power-on self-tests
2. Initially detects and sets up hardware (such as PNP)
3. Locates the MBR on the bootable device
4. Reads the MBR from the bootable device
5. BIOS starts the boot manager
6. Boot Manager (such as GRUB) loads the Kernel and *initrd* (Initial RAM Disk)
7. Boot Manager starts the Kernel
8. Kernel decompresses itself and then takes control of the booting of the system
9. **Note:** If the *initrd* exists, it is mounted as the root of the system. The linuxrc script then loads the drivers to access the real system root and *initrd* is unmounted.
10. Kernel mounts the real system root
11. Kernel checks and sets up the console, reads BIOS settings and initializes basic hardware interfaces
12. Drivers probe and initialize existing hardware
13. Kernel now controls the system, including hardware access and allocation of CPU time and memory
14. Kernel loads the init program, which becomes PID 1
15. Then init calls the */etc/init.d/boot* to activate swap, disk quotas, logical volumes and software RAID and finally mount local file systems from the */etc/fstab*

Figure C-13

16. Next init calls the `/etc/init.d/rc` script, which switches the system to the default runlevel defined in the `/etc/inittab`
17. The runlevel scripts are initialized, and if the default runlevel includes multiuser characteristics, the `getty` processes are initialized and users can then log in

Init and Runlevels

This section provides a layout and comparison of the paths and methods that are different between Red Hat and SUSE boot processes and runlevels.

The way that Red Hat and SUSE Linux Enterprise Server store and manage their daemons are slightly different. The main path difference is that Red Hat uses and documents its service and runlevel root in the `/etc/rc.d` tree while SUSE Linux Enterprise Server uses the more traditional `/etc/init.d` tree.

SUSE Linux Enterprise Server provides a symbolic link named `/etc/rc.d` that points to the `/etc/init.d` directory for Red Hat compatibility. You'll find upon inspection that Red Hat systems include a symlink named `/etc/init.d` that really points to the directory `/etc/rc.d/init.d`.

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Init files Root	<code>/etc/rc.d/init.d</code>	<code>/etc/init.d</code>
Runlevel Directories	<code>/etc/rc.d/init.d/rcX.d</code>	<code>/etc/init.d/rcX.d</code>
Service Scripts	<code>/etc/init.d/service start/stop</code> <code>service <name> start/stop</code>	<code>/etc/init.d/service start/stop</code> <code>rc<name> start/stop</code> (executes <code>/sbin/rcname scripts</code>)
Service Commands	<code>chkconfig --level <name> on/off</code>	<code>chkconfig --level <name> on/off</code> <code>insserv <name></code>

SUSE Linux Enterprise Server Boot Options

The intent of this section is to have a quick reference for the multitude of options that can be specified on the boot prompt for a SUSE Linux Enterprise Server system. In most cases the option text is followed by a "=" sign and then the value, such as: `Language=en`.

Option name	Description	Example
Language	set the language	<code>de_DE</code>
Keytable	load this keytable	<code>de-lat1-nd</code>
Display	set the menu color scheme	<code>Color Mono Alt</code>
Install	URL to installation source media	<code>http://server/path/to/source</code>
Instmode	Install mode	<code>cd hd nfs smb ftp http tftp</code>

Figure C-14

14

Option name	Description	Example
HostIP	Client IP address	10.0.0.1
Netmask	Client netmask	255.0.0.0
Gateway	Client gateway	10.0.0.254
Server	Installation server address (deprecated, use install= instead)	10.0.0.2
Nameserver	DNS server	10.0.0.3
Proxy	FTP or HTTP proxy server	10.0.0.4
Proxy port	Proxy server port	8000
Partition	Path to local install source	<i>Partition=hda1</i>
Server dir	Path relative to "Server" param (deprecated, use install= instead)	<i>/path/to/source</i>
Netdevice	Network interface to use during install	eth0
BOOTPWait	Sleep x seconds between network activation and starting bootp	5
BOOTPTimeout	x second time-out for BOOTP requests	10
DHCPTIMEout	x second time-out for DHCP requests	60
TFTPTIMEout	x second time-out for TFTP connection	10
ForceRootimage	Load the installation system into RAM disk	0 or 1
Textmode	Start YaST in text mode (for Vmware, older monitors etc)	0 or 1
Username	Username for install server (i.e. FTP)	User
Password	Password for install server (e.g., FTP)	<i>Password=novell</i>
WorkDomain	Domain/workgroup name for Samba install	<i>Workgroup=TUXNET</i>
Forcelnsmod	For insmod to run with -f	0 or 1
DHCP	Start DHCP daemon now, but see UseDHCP	0 or 1
UseDHCP	Use DHCP instead of BOOTP (DHCP is default)	0 or 1
MemLimit	Ask for swap if free memory drops below xxx kB	1000
Manual	Start linuxrc in manual mode	0 or 1
NoPCMCIA	Don't start PCMCIA	0 or 1
AutoYaST	Path to AY control file	<i>autoyast=http://server/control.xml</i>

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-15

Option name	Description	Example
Vnc	Install over VNC instead of local console	0 or 1
vnc_password	Set VNC password (min. 5 chars)	<i>vnc_password=novell</i>
UseSSH	Install over SSH instead of local console	0 or 1
SSHPassword	Set SSH password	<i>SSH Password=novell</i>
insmod	Load xyz kernel mod	<i>insmod=pcnet32</i>
brokenmodules	Modules to ignore or not load	<i>brokenmodules=ig3</i>
acpi	Turns on/off ACPI Options	<i>acpi=off</i>
apic	Clustering Mode support for X64	<i>apic=noapic</i> <i>apic=nolapic</i>
apm	Advanced Power Management	<i>apm=off</i>
barrier	Journaling File System Support for transactional writes	<i>barrier=flush (for reiserfs)</i> <i>barrier=1 (for ext3)</i>
netwait	Causes the network to wait N seconds (for other modules)	<i>netwait=10</i>
LogHost	Log install messages to a remote syslog server in addition to /var/log/YaST2/y2log	<i>LogHost=10.0.0.1 (IP address of logging server)</i>

SERVICES AND DAEMONS

Although not all-inclusive, the following table contains the majority of the situations in which Red Hat and SUSE Linux Enterprise Server differ.

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Apache		
Configuration Files	<i>/etc/httpd/conf/httpd.conf</i>	<i>/etc/apache2/httpd.conf</i>
Include Files and/or Directories	<i>/etc/httpd/conf.d/*.conf</i>	<i>/ec/apache2/conf.d</i> <i>/etc/apache2/sysconfig.d</i> <i>/etc/apache2/vhosts.d</i>
Log Files	<i>/var/log/httpd/*</i>	<i>/var/log/apache2</i>
WebRoot	<i>/var/www/html</i>	<i>/srv/www/htdocs</i>
CGI-BIN Directory	<i>/var/www/cgi-bin</i>	<i>/srv/www/cgi-bin</i>
Commands	<i>/etc/init.d/httpd</i>	<i>/etc/init.d/apache2</i>

Figure C-16

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
BIND/DNS		
Chrooted Directory	<i>/var/named/chroot</i>	<i>/var/lib/named</i>
Files	<i>/etc/named.conf</i> (symlinked to <i>/var/named/chroot/etc/named.conf</i>)	<i>/etc/named.conf</i> (copied from <i>/var/lib/named/etc</i> every time <i>named</i> is started)
	<i>/etc/named.custom</i>	<i>/var/lib/named/etc/named.conf.include</i> (copied from <i>/etc/named.conf.include</i>)
	<i>/etc/rndc.key</i> (symlinked to <i>/var/named/chroot/etc/rndc.key</i>)	<i>/etc/named/conf.include</i> (copied from <i>/var/lib/named/etc</i>) <i>/var/lib/named/*</i> <i>/var/lib/named/etc/named.d</i> <i>/var/lib/named/etc/rndc.key</i>
Zone Files	<i>/var/named/chroot/var/named/*</i>	<i>/var/lib/named/master</i> <i>/var/lib/named/slave</i>
Config Files	<i>/etc/dhcpd.conf</i> (not chrooted by default)	<i>/etc/dhcpd.config</i> (copied from <i>/var/lib/dhcp/etc/dhcpd.conf</i>) (chrooted to <i>/var/lib/dhcp</i>)
DHCPD		
Config Files	<i>/etc/dhcpd.conf</i>	<i>/var/lib/dhcp/etc/dhcpd.conf</i> (copied from <i>/etc/dhcpd.conf</i> every time <i>dhcpd</i> is started)
Client	<i>dhclient</i>	<i>Dhcpd</i>

Figure C-17

17

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
OpenLDAP		
Schema Files	<i>Nis.schema</i> (objectClass posixAccount is Structural)	<i>Rfc2307bis.schema</i> User groups: <i>objectClass</i> <i>posixAccount</i> is auxiliary ObjectClass namedObject is Structural
Samba		
User DB	<u>/etc/samba/smbpasswd</u>	<u>/etc/samba/smbpasswd</u> (default even if OpenLDAP is the user database) Can store samba data in OpenLDAP directory
E-mail		
Default MTA	Sendmail	Postfix
Default IMAP	Dovecot (Cyrus Optional)	Cyrus IMAPd
Default POP3	Dovecot	Qpopper
SendMail		
Package	sendmail and sendmail.cf	Sendmail
Config Files	<u>/etc/mail</u>	<u>/etc/mail</u>
Binaries	<u>/usr/bin</u>	<u>/usr/sbin</u>
NFS		
Server	NFS V4 (Red Hat Enterprise Linux 4 Only) (Fully back-compatible with NFS V3)	NFS V3
Client	NFS V4 (Red Hat Enterprise Linux 4 Only) (Fully back-compatible with NFS V3)	NFS V3
OpenSLP		
Availability	N/A	Yes

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Figure C-18

Network Configuration

The network-configuration-file differences between the two distributions are fairly dramatic. The most obvious is that Red Hat has a sysconfig file named *network* that has no equivalent in SUSE Linux Enterprise Server. The settings in Red Hat's *network* file are split amongst various files in the SUSE Linux Enterprise Server *network* directory.

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Network Configuration		
Sysconfig File	/etc/sysconfig/network	/etc/sysconfig/network/*
File Root	/etc/sysconfig/network-scripts	/etc/sysconfig/network/*
Interface Files	/etc/sysconfig/network-scripts/ifcfg-ethX	/etc/sysconfig/network-scripts/ifcfg-eth-id-MA:CA:DD:RE:SS
Commands	ifconfig	ifconfig
	route	route
	ifup	ifup
	Ifdown	ifdown
		ifstatus
Daemons	/etc/init.d/network	/etc/init.d/network

User Management and Environment

The majority of the files and commands concerning user management are similar; the primary differences are in the GUI/TUI and command-line tools available.

Command-line User Management

The command-line tools are the most similar; the only notable difference is the inclusion of the *sux* command for SUSE Linux Enterprise Server and the ways the distributions create a user home directory and default group.

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
User Management (Command-line)		
Adding Users	useradd	useradd
	- creates home dir by default	- must use -m to create home directory
	- creates private group by default	- uses public group "users" by default

Figure C-19

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Privilege Elevation	<i>su</i> <i>su <CR></i> (non-login root) <i>su - <CR></i> (login root) <i>su -c</i> "cmd to run"	<i>su</i> / <i>sux</i> * <i>su <CR></i> (non-login root) <i>su - <CR></i> (login root) <i>su -c</i> "cmd to run"
Note: The <i>sux</i> command shares the same syntax as the <i>su</i> command and exports the <i>xhost</i> and <i>DISPLAY</i> options when elevating to root.		
Database Files (and Permissions)	<i>/etc/passwd</i> 644 <i>/etc/shadow</i> 400 <i>/etc/group</i> 644	<i>/etc/passwd</i> 644 <i>/etc/shadow</i> 640 <i>/etc/group</i> 644
Encryption	MD5 (default)	DES (default) DES MD5 Blowfish
Additional Authentication Support	Kerberos SMB LDAP WinBind NIS HESIOD	Kerberos SMB LDAP WinBind NIS

Graphical User Management

The differences between Red Hat and SUSE Linux Enterprise Server GUI and TUI tools for user management are more than slightly divergent. While both have a single portal point for creating users and groups, Red Hat uses a single application and SUSE Linux Enterprise Server uses a module that is part of the YaST management system.

Action	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
User Management (GUI)		
Adding Users	<i>system-config-users</i> 1. Click on Add User 2. Fill in the Create New User Dialog text boxes (limited to: User Name, Full Name, Password, Shell, Create Home Dir, Private Group, Manual UID)	<i>YaST -> Security and Users -> Edit and Create Users</i> 1. Click on the Add button 2. Fill in the User Data Tab text boxes Tab # 1 includes: Full User Name, User Name, Password, Disable User Login

Figure C-20

Action	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
		Tab # 2 includes: UID, Home Dir, Add'l Info, Login Shell, Default Group, Groups Tab # 3 includes: Day before Expire, Grace Login, Max Days, Min Days, Expire Date
Privilege Elevation	N/A	<i>kdesu</i> - runs the target command as the root user - exports the xhost and DISPLAY settings - run under X as a non-root user
Database Files	<i>/etc/passwd</i> 644 <i>/etc/shadow</i> 400 <i>/etc/group</i> 644	<i>/etc/passwd</i> 644 <i>/etc/shadow</i> 640 <i>/etc/group</i> 644
Encryption	MD5 (default)	DES (default) DES MD5 Blowfish
Additional Authentication Support	Kerberos SMB LDAP WinBind NIS HESIOD	Kerberos SMB LDAP WinBind NIS

User Environment

The files that make up a user's environment are roughly similar on the two systems, although Red Hat Enterprise Linux has a few more files by default, as shown below. Additionally, SUSE Linux Enterprise Server employs a few .local files that aren't normally used on Red Hat systems.

File	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
Global Profile	<i>/etc/profile</i>	<i>/etc/profile</i>
Local Profile	<i>~/.bash_profile</i>	
Global Bash	<i>/etc/bashrc</i>	<i>/etc/bash.bashrc</i>
Local Bash	<i>~/.bashrc</i>	
History	<i>~/.bash_history</i>	<i>~/.bash_history</i>
Logout	<i>~/.bash_logout</i>	

Figure C-21

21

SUSE Linux Enterprise Server does not include all the files that Red Hat Enterprise Linux does, but if present, the files are read in the expected order. Be certain to use the various `.local` files—typically used in the user home directories (such as the `~/bashrc.local` file, which would not be updated by any security packages)—for storing user aliases, functions and anything else you might need.

SYSTEM ADMINISTRATION

System-administration tools on the two systems are vastly different. If you're accustomed to the multitude of tools in Red Hat, you'll enjoy the SUSE Linux Enterprise Server common administration interface called YaST, which stands for "Yet Another Setup Tool."

Instead of requiring administrators to install and use many different interfaces for administering applications, YaST is a "one-stop shop" that allows for either text-menu-based or full graphical administration of SUSE Linux Enterprise Server. Below are some of the more common tasks/services that can be managed via YaST:

- Adding, deleting or updating software
- Managing users and groups and security settings
- Administering installed services (Apache, DNS, DHCP, TFTP, etc.)
- Offering deployment services
- Configuration and build management through AutoYaST
- Automatic installation server setup and configuration
- Installing and configuring hardware
- Managing disks and volumes

SuSEconfig—The Normalizer

While each YaST module/component performs slightly different tasks based on the application or service being administered, it typically modifies both application configuration files (e.g., `/etc/dhcpd.conf`) as well as any related files in `/etc/sysconfig` (e.g., `/etc/sysconfig/dhcpd`). After making the needed changes, YaST runs a bash shell script called `SuSEconfig` (located in `/sbin`), which performs several key tasks. `SuSEconfig` configures the system according to the variables that are set in the various `/etc/sysconfig/` files either by YaST or by hand. It uses the subsystem-specific scripts in `/sbin/conf.d/` to configure the various subsystems. For example, the variables in `/etc/sysconfig/postfix` are evaluated by the script `/sbin/conf.d/SuSEconfig.postfix`.

Note: `SuSEconfig` is an excellent tool for those managing larger sets of system because it normalizes how changes are made to the system's running services and can assist in reducing downtime due to improperly applied changes or missed steps in complex configuration changes.

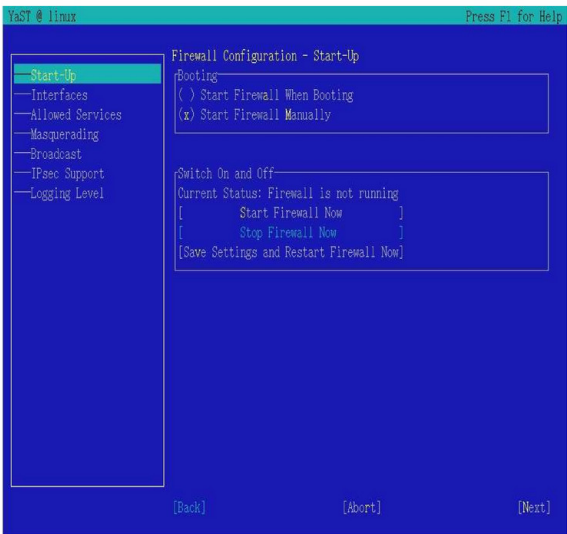
YaST The GUI Version

While the command line holds a certain *savoir-faire* or "geek chic," most day-to-day operations are performed by those who rely on the menus and wizards of GUI tools for system

Figure C-22

administration. Most experienced system administrators have managed to misconfigure a service by leaving an important file untouched or forgetting to do a supporting task, which then causes an important service to malfunction. Using YaST (instead of the command line) and a file editor to configure services organizes and presents all the necessary files in a single set of wizard-like interface steps. This greatly reduces the likelihood of misconfiguration.

To run the GUI version of YaST, select the YaST icon on the start bar on the KDE desktop, press *Alt-F2* in KDE or GNOME, and type "*yast2*" or click the System Tools -> System Configuration menu entry on the GNOME task bar. Once inside YaST, there is a scope pane on the left side that contains major groupings of modules. Upon your selecting an icon on the left, the modules contained in that area or category appear to the right. Selecting a module starts the module interface, and configuration is done in that interface. This can and often does integrate changes several text-configuration files.



YaST— the Text Mode Version

YaST, the GUI version (*/sbin/yast2*), has a complete shadow TUI version complement (*/sbin/yast*) that uses the same interface style and groupings. Literally, everything is the same but rendered in the Ncurses library routines, similar to the old *syscon* (from NetWare) utility. Instead of clicking on things with the mouse to select items and configure text boxes, use the

Figure C-23

Tab and arrow keys for navigation. The interface has assigned Alt-key combinations to make things very quick and easy to choose. For instance, configuring a typical network service in the GUI YaST involves a number of clicks and typing, but in the TUI mode, it is very quick; you'll never need to move your hands from the keyboard.

YaST the TUI version can be invoked quickly and easily from the command line, and typically you use the TUI version when you happen to be at the command line anyway, and it is much easier to just invoke the `yast` command and supply the needed module as a parameter.

For example, to load the firewall-setting module directly and not have to navigate from YaST -> Security and Users -> Firewall, you could just type the command: `# yast firewall`

This brings up the screen shown below:

YaST—the Text-mode Version

While the modules will be used primarily as parameters with the TUI version of YaST, they do work with the GUI version, such as when you're running `/sbin/yast2` from a Run dialog. The YaST TUI modules list is a handy reference to have, particularly if you spend a lot of time doing remote console work via `ssh`. The complete list is obtainable by typing "`yast -l`", with the output being somewhat lengthy. The correct syntax for specifying a module with the `yast` or `yast2` command is `/sbin/yast module`

Rather than explaining every module, the "Comparing System Administration Tools" section below maps the various `/sbin/yast` modules along with the GUI YaST equivalents to Red Hat's `system-config-` suite of various configuration tools.

Comparing System Administration Tools

Red Hat uses a multitude of lengthily-named tools, while SUSE Linux Enterprise Server uses YaST—either the TUI version (`/sbin/yast`) or the GUI version (`/sbin/yast2`)—as a single portal for management. The following table compares the Red Hat tool with its YaST-module counterpart in both GUI and TUI modes.

Red Hat Enterprise Linux Utility	SUSE Linux Enterprise Server YaST Module
system-config-authentication	YaST -> Security Settings <code>/sbin/yast security</code>
system-config-boot	YaST -> System -> Boot Loader Configuration <code>/sbin/yast bootloader</code>
system-config-date	YaST -> System -> Date and Time
system-config-display	SaX2 YaST -> Hardware -> Graphics Card and Monitor
system-config-httpd	YaST -> Network Services -> Web Server

Figure C-24

Red Hat Enterprise Linux Utility	SUSE Linux Enterprise Server YaST Module
system-config-keyboard	YaST -> Hardware -> Keyboard Layout
system-config-kickstart	YaST -> Software -> Installation Server
system-config-language	YaST -> System -> Language Selection
system-config-lvm	YaST -> System -> LVM
system-config-mouse	YaST -> Hardware -> Mouse Model
system-config-netboot	N/A
system-config-network	YaST -> Network Devices
system-config-network-gui	YaST -> Network Devices
system-config-network-tui	/sbin/yast lan
system-config-nfs	YaST -> Network Services -> NFS Server YaST -> Network Services -> NFS Client
system-config-packages	YaST -> Software -> Install and Remove Software
system-config-printer	YaST -> Hardware -> Printer
- system-config-printer-gui	YaST -> Hardware -> Printer
- system-config-printer-tui	/sbin/yast printer
system-config-rootpassword	passwd root
system-config-samba	YaST -> Network Services -> Samba Server
system-config-securitylevel	YaST -> Security and Users -> Firewall
system-config-securitylevel-tui	/sbin/yast security
system-config-services	YaST -> System -> System Services (Runlevel)
system-config-soundcard	YaST -> Hardware -> Sound
system-config-time	YaST -> System -> Date and Time
(same as system-config-date)	
system-config-users	YaST -> Users and Groups

SERVICE AND SUPPORT PACK FROM SUSE VS. RED HAT UPDATES

Red Hat Enterprise Linux and SUSE Linux Enterprise Server both use an online update mechanism: respectively, Red Hat Network and either YaST Online Update (YOU) for internal software distribution or ZENworks Linux Management for updates from Novell.

Figure C-25

25

Enabling Red Hat Network Support

Typically, a system administrator will arrange for a support subscription from Red Hat, register the system being supported via the `rhnc_register` command and then connect via the `up2date` or `rhnc_applet-gui/rhnc_applet-tui` utilities. A wizard then outlines the remaining process. A visit to the `rhnc.redhat.com` site is often needed to enable the support subscription if the registration process is unsuccessful.

After enabling the support, the system can be updated manually, either with the `up2date` command-line tool or with the Red Hat Network Alert Notification Tool from the GUI desktop.

Enabling SUSE Linux Enterprise Server Support

The YOU (YaST Online Update) Server GUI/TUI client is extremely easy to use, as the root user selects YaST -> Software -> Online Update. The system will fetch the list of update servers and present a dialog that lets the system administrator configure how updates are performed.

The YaST Online Update client allows for automatic updating on a scheduled basis; just click the Configure Fully Automatic Update button and configure it to update at a particular time. You can also configure it to download only patches and not other items.

SUSE Linux Enterprise Server also offers the `online_update` command line tool to automate the updating of systems.

A note about updating kernels: Red Hat Enterprise Linux has traditionally updated the system with the new kernel and added entries in the boot menu (GRUB or LILO), while SUSE Linux Enterprise Server has traditionally replaced the current kernel entry with the new one. SUSE Linux Enterprise Server will allow for keeping entries after the application of Support Pack 2.

FILE-SYSTEM HIERARCHY

Both SUSE Linux Enterprise Server and Red Hat Enterprise Linux conform to the Linux Standards Base (LSB) specification, which now includes the Filesystem Hierarchy Standard (FHS). Red Hat Enterprise Linux 3 conforms to LSB 1.3, and SUSE Linux Enterprise Server 9 conforms to LSB 2.0. Supporting the LSB minimizes the differences between the two distributions filesystem layouts, and system administrators can typically rely on configurations and files being in common directory locations.

You can find the LSB and FHS specifications along with a number of other free standards at the Free Standards site: <http://freestandards.org/>

One of the main differences that existed in previous versions was that Red Hat Enterprise Linux used the `/mnt` directory for all mounting of file systems, floppies and CD-ROMs, including USB keys and network file systems. SUSE Linux Enterprise Server has long used the `/mnt` directory as a container for network-based file systems, while the `/media` directory is for any attached but not integrated media object. For example, if you were to attach a USB key drive to a SUSE Linux Enterprise Server system, it will mount the USB key to a directory that is autogenerated by

Figure C-26

the system in the `/media` directory, such as `/media/CRUZER` for a Sandisk Cruiser 1 GB USB drive.

With Red Hat's Enterprise Linux Version 4 release, both similarly use the `/mnt` and `/media` directories, thus reducing a point of possible confusion.

	Red Hat Enterprise Linux	SUSE Linux Enterprise Server
File Systems and Volumes		
Mounting	Removable media <ul style="list-style-type: none">- <code>udev</code> (Red Hat Enterprise Linux 4)- <code>hotplug+updfstab</code> (Red Hat Enterprise Linux 3)	Removable media <ul style="list-style-type: none">- <code>submount2+subfs</code>- <code>/dev/cdrom</code> and <code>/media/cdrom</code> are governed by <code>subfs</code>
Directories	<code>/mnt</code> <code>/media</code>	<code>/media</code>
Permissions		Enables POSIX ACL's by default
Default FS	Ext3	Reiser
File Systems Not Supported	Reiser* JFS and XFS not supported in the kernel or as modules	N/A
Note: *Reiser needs special driver support in Red Hat Enterprise Linux 4. Use the "linux reiserfs" load-time option to enable.		

DOCUMENTATION AND HELP RESOURCES

SUSE Linux Enterprise Server documentation is stored in the `/usr/share/doc` folders. The package's directory contains package-specific documentation. The SUSE Linux Enterprise Server administration guide (PDFs and HTML) are located under `manual/sles-admin_en`, and release notes are contained in `release-notes/`.

Man pages: In accordance with the default manpage directory specified in the FHS (www.pathname.com/fhs/pub/fhs-2.3.html#USRSHAREMANMANUALPAGES), additional man-page search directories are added by default to the `MANPATH` variable. This, of course, depends on the software installed, but the most common additions are:

Figure C-27

27

- `/usr/local/man` - Added for convenience. Typically, custom-built OSS packages will use `/usr/local` as an install path, in which case man pages would appear in `/usr/local/man` instead of `/usr/share/man`
- `/usr/X11R6/man` - X windows- `/opt/gnome/share/man` - Home to (some) gnome man pages

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

Index

Symbols

#include 7-7

/etc/aliases 6-8, 6-42

/etc/apache2/ 6-53

/etc/apache2/default-server.conf 6-55

/etc/apache2/vhosts.d/ 6-58

/etc/apparmor.d/ 7-5, 7-11

/etc/apparmor.d/abstractions/ 7-9

/etc/apparmor.d/tunables/ 7-9

/etc/apparmor/abstractions/ 7-7

/etc/apparmor/profiles/extras/ 7-11

/etc/default/passwd 2-35

/etc/default/useradd 2-32

/etc/fstab 4-17, 4-26

/etc/fstab and iSCSI 9-22

/etc/fstab, options 4-29

/etc/group 2-29

/etc/grub.conf 5-18

/etc/HOSTNAME 3-32

/etc/ietd.conf 9-11

/etc/init.d/ 5-29

/etc/init.d/boot 5-6

/etc/init.d/boot.apparmor 7-5, 7-26

/etc/init.d/rc 5-29

/etc/init.d/rcx.d/ 5-28

/etc/initiatorname.conf 9-20

/etc/inittab 5-6, 5-23, 5-25

/etc/insserv.conf 5-36

/etc/iscsi.conf 9-19

/etc/login.defs 2-35

/etc/ocfs2/cluster.conf 10-6

/etc/passwd 1-45, 2-29

/etc/permissions.* 2-42

/etc/postfix/ 6-8

/etc/postfix/access 6-31

/etc/postfix/main.cf 6-15, 6-21

/etc/postfix/master.cf 6-10

/etc/resolv.conf 3-32

/etc/security/pam_pwcheck.conf 2-35

/etc/shadow 1-45, 2-29

/etc/sysconfig/hardware/ 3-25

/etc/sysconfig/kernel 5-5

/etc/sysconfig/mail 6-15

/etc/sysconfig/network/ 3-22

/etc/sysconfig/network/config 3-24

/etc/sysconfig/network/ifcfg.template 3-25

/etc/sysconfig/network/routes 3-29

/etc/sysconfig/postfix 6-15

/etc/sysconfig/xendomains 8-26

/etc/xen/ 8-19

/etc/xen/auto/ 8-26

/root/autoinst.xml 1-49

/sbin/hwup 3-25

/sbin/init 5-4–5-5, 5-23

/sbin/rcapparmor 7-26

/srv/www/htdocs/ 6-51

/sys/kernel/security/apparmor/profiles 7-28

/var/lib/open-iscsi/ 9-19

/var/log/audit/audit.log 7-13

/var/log/wtmp 2-40
/var/spool/postfix/ 6-8

A

aamatch_pcre kernel module 7-5
access lookup table 6-31
ACPI 1-3
Add Profile Wizard 7-18
aliases lookup table 6-42
Allow 7-15
AMD Pacifica 8-3
anaconda A-2
Apache 6-48
Apache configuration 6-53
Apache, installation 6-49
APIC 1-4
AppArmor 7-1, 7-5
apparmor kernel module 7-5
AppArmor profiles 7-7–7-8
AppArmor profiles, administration 7-11
AppArmor profiles, permissions 7-10
AppArmor rules 7-8
AppArmor, monitoring 7-31
AppArmor, starting and stopping 7-26
AppArmor, status 7-27
autodep 7-18, 7-21
automated installation A-1, A-16
automounter 1-47
AutoYaST A-1
Autoyast 1-49

B

backup strategy 4-2
BIOS setup 1-2

block bitmap 4-6
block group 4-5
Blowfish 1-31, 2-36
boot loader 5-7
boot loader, first stage 5-8
boot loader, second stage 5-8
boot manager 5-7
brctl 8-33
bridge 8-29
broadcast address 3-17
browser 6-48

C

CA 1-42
CA management 1-42
canonical lookup table 6-33
capability, POSIX 7-8
Certification Intro-3
certification authority 1-42
chkconfig 5-37
CLE 10 Intro-1
cluster file system 10-1
complain 7-6, 7-19
configuration, network 1-33
Ctrl+Alt+Del 2-37

D

dd 4-46
default gateway 3-9
default route 3-27, 3-29
delete a profile 7-21
demilitatized zone 3-10
Deny 7-15
dependency 2-12

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

DES 2-36
device attributes 3-17
DHCP 1-36, 3-5–3-6
directives 6-54
discovery.db 9-19
Distributed Replicated Block Device 10-4
DMA 1-4
DNS 3-6
DocumentRoot 6-51
dom0_mem 8-9
domain 3-7
domain migration 8-35
domain name 3-7
domain search list 3-8
domain0 8-6
domainU 8-7
DRBD 10-4
dump 4-27
dumpe2fs 4-34

E

e2fsck 4-33
Edit 7-16
enforce 7-6, 7-20
error.conf 6-53
eth0 3-16
ethernet 3-4
Ethernet adapter 3-16
Ethernet devices 3-16
ethtool 3-24
exec 4-17
Exercise Conventions Intro-9
ext2 4-3
ext3 4-3
extended partition 1-11, 1-21

external zone 3-10

F

fdisk 4-13, 4-20
file system 4-2
file system check 4-27
file system type 4-27
file system, create 4-21
file system, encryption 4-22
file systems, documentation 4-12
file systems, internals 4-4
files, network configuration 3-22
firewall 1-32, 3-10
firewall zone 3-10
firewall, interface 3-10
format a partition 4-21
FQDN 3-7
fsck 4-32
fuser 4-30

G

gateway 3-9
genprof 7-21
GFS 10-1–10-2
GID 2-19
Glob 7-16
Glob w/Ext. 7-16
Global File System 10-2
Grand Unified Boot Loader 5-7
graphic card 1-47
group account 2-18
group administration 2-19
group descriptor 4-6
Group ID 2-19

group, default 2-31
group, edit 2-28
group, new 2-28
group, secondary 2-31
GRUB 5-7
GRUB shell 5-10
grub-md5-crypt 5-20

H

halt 5-31, 5-42
hardware address 3-16
hardware clock 1-8
hardware RAID 4-49
home directory 2-23
host 3-8
host name configuration 3-32
hostname 1-29, 3-6–3-7
hotplug 3-11, 9-22
HTTP 6-48
httpd.conf 6-53
hwup 3-25
Hyper Text Transfer Protocol 6-48
Hypervisor 8-6

I

IDE controller 4-14
ietd 9-11
ifdown 3-25
ifup 3-2, 3-25
Inherit 7-14
init 5-23, 5-41
init 0 5-42
init 6 5-42
init q 5-27

init=/bin/bash 5-19
initramfs 5-4
initrd 5-4
inode bitmap 4-6
inode table 4-7
insserv 5-35
installation 1-2
installation proposal 1-9
installation server A-5
installation settings 1-9
installation source, management 2-8
installation, automated A-1
installation, configuration 1-29
installation, network 1-31
installation, partitioning 1-10
installation, start 1-28
installation, troubleshooting 1-50
Intel Vanderpool 8-3
interface 3-2, 3-4
internal zone 3-10
internet connection, test 1-38
Internet Small Computer Systems Interface 9-1
ip 3-11, 3-14
IP address 3-6
ip address add 3-20
IP address auto configuration 3-27
ip address del 3-21
ip address show 3-15
IP address, dynamic 3-5
IP address, static 3-5
IP forwarding 3-9
ip link set 3-21
ip link show 3-17
ip route add 3-28
ip route delete 3-29
ip route show 3-27

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

IPv4 3-5, 3-16

IPv6 3-17

iqn 9-9

iSCSI 9-1

iSCSI authentication 9-7

iSCSI configuration 9-4

iSCSI Enterprise Target Daemon 9-11

iSCSI initiator 9-2

iSCSI qualified name 9-9

iSCSI target 9-2

iscsiadm 9-19

iscsid 9-20

J

journaling 4-11

K

kernel module 3-3–3-4

kickstart A-2

L

LABEL 4-26

language 1-5

last 2-40

LDAP 1-42

LDAP client 1-46

LDAP Server. address 1-46

LILO 4-15, 5-9

LILO (LInux LOader) 5-8

listen.conf 6-53

local time 1-8

local users 1-45

locate 2-42

locatedb 2-42

logical partition 1-11, 1-19

logical volume 4-38

login shell 2-23

login, delay 2-39

logprof 7-23

lookup tables 6-30

loopback device 3-11, 3-16

lost+found 4-32

lsscsi 9-21

lvcreate 4-48

lvextend 4-48

LVM 4-1, 4-37

LVM, features 4-39

lvreduce 4-48

lvscan 4-48

M

MAC 7-1

mail envelope 6-33

mail header 6-33

mail, forward 6-24

mailq 6-46

Maintenance Intro-5

Mandatory Access Control 7-1

Manually Add Profile 7-17

master daemon 6-9

Maximum Transfer Unit 3-11

MD5 2-36

memory test 1-4

metadata 4-11

mke2fs 4-23

mkfs 4-13, 4-23

mkfs.ext3 4-23

mkinitrd 5-5

mkreiserfs 4-23–4-24
mount 4-25, 4-28
mount options 4-27
mount point 4-23, 4-27
MTU 3-11, 3-24
multitasking 2-18
multiuser 2-18

N

name resolution 3-32
name server 3-6
ncurses, YaST 2-2
network 3-7
network card 3-3–3-4
network configuration 3-1
network configuration, manual 3-14
network file systems 4-10
network interface 1-33, 3-2
network interface, setup 3-14
network mask 3-6, 3-17
network, configuration 1-31
NetworkManager 3-2, 3-35
New Profile Wizard 7-12
newaliases 6-44, 6-46
nm-applet 3-36
nm-tools 3-35
noauto 4-17
node.db 9-19
Novell CLE 10 Intro-1
Novell Customer Center Intro-6, 1-39, 1-41,
2-13

O

o2cb 10-8

o2cb configure 10-8
OCFS 10-1
OCFS2 10-3
ocfs2 10-8
ocfs2console 10-10
OCSF2 configuration 10-4
Online Resources Intro-7
Online Update 1-40
online update 2-14
open relay 6-27
Oracle Cluster File System 2 10-3

P

package groups 2-10
para-virtualization 8-5
partition table 4-13
partition, delete 1-22
partition, edit 1-22
partition, new 1-18
partition, resize 1-23
partitioning 4-19
partitioning scheme 4-15
partitions 1-11, 4-13
partitions, naming convention 4-14
partprobe 4-20
password 2-18
password age 2-36
password checks 2-36
password encryption 2-36
password expiration 2-25, 2-37
password, root 1-30
pattern 1-26
patterns 2-10
permissions 2-42
peth 8-32

1 **HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED**

physical extent 4-41
physical volume 4-38
POSIX capabilities 7-8
postalias 6-44, 6-46
postcat 6-46
postconf 6-23, 6-46
postdrop 6-4, 6-46
Postfix 6-2
postfix 6-46
Postfix master daemon 6-9
Postfix tools 6-46
Postfix, modularization 6-3
postmap 6-30, 6-46
postsuper 6-47
poweroff 5-42
Power-On Self Test (POST), 5-7
primary partition 1-11
printer 1-47
Profile 7-14
profile, delete 7-21
profile, new 7-12
profile, update 7-18
profiles, reload 7-29
promiscuous mode 3-18
pvcreate 4-47
pvmove 4-47
pvscan 4-47
PXE A-1

Q

QT, YaST 2-2
queue manager 6-6

R

RAID 4-49
RAID 0 4-49
RAID 1 4-49
RAID 5 4-50
RAID 6 4-50
RBL 6-19
rc 5-31
rcapparmor 7-26
Realtime Blackhole List 6-19
reboot 5-31, 5-42
recipient_canonical lookup table 6-34
ReiserFS 4-3, 4-7
ReiserFS format 4-8
reiserfsck 4-33
reiserfstune 4-35
relayhost 6-25
relocated lookup table 6-37
Rescue System 1-4
resize_reiserfs 4-35
resize2fs 4-35
root partition 1-12
root password 1-30
route 3-8
router 3-6
routing 3-8, 3-26
routing table 3-9, 3-26
RPM 1-26
RPM package 1-26
runlevel 5-23
runlevel (command) 5-24
runlevel, change 5-41

S

SCSI 4-13, 9-2
security patch 2-12
security settings, local 2-33
SELinux 7-1
sender_canonical lookup table 6-36
Sendmail 6-2
server-tuning.conf 6-53
sit0 3-16
skeleton directory 2-31
smtpd 6-5
software management 2-8
software RAID 4-1, 4-49
software update 2-12
software, installation 2-9
software, search for 2-11
sound 1-47
SSH 1-32
ssh-copy-id 10-5
ssh-keygen 10-5
SSL 1-42
ssl-global.conf 6-54
Subdomain 7-6
subnet mask 3-6
superblock 4-6
Support Intro-5
SuSEconfig 2-5
swap partition 1-12
SysRq keys 2-43

T

time zone 1-8
TLS 1-42
transport lookup table 6-39

tune2fs 4-34

U

UCE 6-19
UID 2-18
UID 0 2-18
uid.conf 6-53
umount 4-30
unconfined 7-14, 7-29
Unique Universal ID 9-22
unsolicited commercial email 6-19
update 2-12
Update Profile Wizard 7-18
updatedb 2-42
user account 2-18
user administration 2-19
user authentication 1-43
User ID 2-18
user root 2-18
user, edit 2-21
user, local 2-20
user, new 2-21
user, system 2-20
useradd 2-32
username 2-18
UTC 1-8
UUID 4-26, 9-22

V

veth 8-32
vgcreate 4-47
vgexpand 4-47
vgreduce 4-47
vgremove 4-47

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED

vhost.d/ 6-53
vhost.template 6-58
vif 8-32
vim 7-24
virtual host 6-57
virtual lookup table 6-41
virtual machine monitor 8-1, 8-6
Virtualization 8-2
virtualization, full 8-4
virtualization, para 8-5
VMware 8-1
volume group 4-38
volume group, name 4-41

W

web browser 6-48
web server 6-48
web server, start 6-49

X

Xen 8-1
Xen architecture 8-6
Xen configuration 8-19
Xen networking 8-28
Xen virtual machine monitor 8-9
xenbr0 8-32
xend 8-7
XFS 4-3
xm 8-21

Y

YaST 2-1
YaST Autoinstallation module A-11

YaST bootloader module 5-13
YaST Control Center 2-5
YaST Expert Partitioner 1-16, 4-18
YaST iSCSI initiator module 9-13
YaST iSCSI target module 9-5
YaST modules 2-45
YaST Online Update 2-12
YaST online update 1-41
YaST Runlevel Editor 5-37
YaST Xen module 8-12
YaST, installation 1-1
YaST, ncurses interface 2-3
YaST, network configuration 3-2
YaST, software 1-25
yast2 lvm_config 4-45
YOU 2-12

1 HARDCOPY PERMITTED-NO DISTRIBUTION ALLOWED